

**SOLUCIÓN DEL PROBLEMA DE RUTEO DE VEHÍCULOS CON DEMANDAS
ESTOCÁSTICAS (VRPSD) MEDIANTE LA METAHEURÍSTICA OPTIMIZACIÓN
POR ESPIRAL**

NATALIA ALEJANDRA GELVES TELLO

RICARDO ANDRÉS MORA MORENO

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE ESTUDIOS INDUSTRIALES Y EMPRESARIALES
BUCARAMANGA**

2015

SOLUCIÓN DEL PROBLEMA DE RUTEO DE VEHÍCULOS CON DEMANDAS ESTOCÁSTICAS (VRPSD) MEDIANTE LA METAHEURÍSTICA OPTIMIZACIÓN POR ESPIRAL

NATALIA ALEJANDRA GELVES TELLO

RICARDO ANDRÉS MORA MORENO

Trabajo de grado para optar al título de Ingeniero Industrial

Director:

HENRY LAMOS DÍAZ

PhD. En Física-matemática

Codirector:

JAVIER ARIAS OSORIO

MSc. En Administración

UNIVERSIDAD INDUSTRIAL DE SANTANDER

FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS

ESCUELA DE ESTUDIOS INDUSTRIALES Y EMPRESARIALES

BUCARAMANGA

2015

DEDICATORIA

De ante mano a Dios, por brindarme la paciencia, sabiduría y fortaleza para lograr mis metas.

A mis padres y hermano que siempre han estado ahí ofreciéndome su amor y apoyo incondicional.

A Eduardo Duarte, Anabel de León, Fernando Tello, capoeira Nativos y demás familiares y amigos que me han apoyado en el transcurso de mi carrera.

A Libia Hernández, Horacio Gelves, Carmen Arocha y Laura Peñaranda, quienes desde el cielo han sido mis guías en cada paso que doy.

Natalia Alejandra Gelves Tello

Primero que todo este trabajo está dedicado a Dios, quien me ha iluminado en cada etapa mi vida y con quien estoy inmensamente agradecido por las grandes bendiciones que me ha dado.

A mis padres Julian Mora y Linda Moreno, quienes han sido mi inspiración diaria para seguir adelante y me han proporcionado la mejor educación y ejemplo.

A mi hermano Julian Eduardo Mora, quien ha sido mi guía, mi modelo a seguir, quien me ha apoyado en todas las decisiones de mi vida.

A mi familia y amigos, quienes a lo largo de mi vida han creído en mis capacidades y me han siempre brindado palabras de apoyo y esperanza.

Ricardo Andrés Mora Moreno

AGRADECIMIENTOS

Al Doctor Henry Lamos Díaz por guiarnos y apoyarnos en el presente proyecto; por su comprensión y confianza ante las distintas situaciones presentadas.

Al magíster Javier Arias Osorio por las asesorías brindadas durante el desarrollo del proyecto.

A la Universidad Industrial de Santander, Escuela de Estudios Industriales y Empresariales y al Grupo OPALO por la formación académica y los conocimientos ofrecidos durante el transcurso de la carrera.

A nuestros compañeros del Grupo OPALO por los consejos y la ayuda brindada para la consecución de este proyecto.

CONTENIDO

	Pág.
INTRODUCCIÓN.....	16
1. PLANTEAMIENTO DEL PROBLEMA	19
2. JUSTIFICACIÓN DEL PROBLEMA.....	21
3. OBJETIVOS.....	23
3.1 OBJETIVO GENERAL.....	23
3.2 OBJETIVOS ESPECÍFICOS	23
4. REVISIÓN DE LA LITERATURA.....	24
5. MARCO CONCEPTUAL	33
5.1 OPTIMIZACIÓN COMBINATORIA.....	33
5.2 COMPLEJIDAD COMPUTACIONAL	34
5.3 MÉTODOS DE SOLUCIÓN.....	37
5.3.1 Métodos Exactos	38
5.3.2 Métodos Aproximados	39
5.3.2.1 Métodos Heurísticos.....	39
5.3.2.2 Métodos Metaheurísticos.....	42
5.4 VEHICLE ROUTING PROBLEM (VRP)	58
5.4.1 Vehicle Routing Problem with Stochastic Demands (VRPSD)	63
5.4.1.1 Políticas de servicio	64
5.4.1.2 Elección de la política de servicio	66
5.4.1.3 Modelamiento matemático del VRPSD	67
6. EXPERIMENTACIÓN.....	69
6.1 FUNCIONES DE PRUEBA	69
6.2 APLICACIÓN DE LA METAHEURÍSTICA OPTIMIZACIÓN POR ESPIRAL A UN PROBLEMA DE TSP	69
7. CÁLCULO DE LA FUNCIÓN OBJETIVO.....	75

8. DEFINICIÓN DEL NÚMERO DE ESCENARIOS.....	78
9. INSTANCIAS	81
10. DESARROLLO DEL ALGORITMO PARA EL VRPSD	84
11. DISEÑO EXPERIMENTAL	92
11.1 DEFINICIÓN DE LOS PARÁMETROS DEL ALGORITMO OPTIMIZACIÓN POR ESPIRAL.....	92
11.2 DEFINICIÓN DEL DISEÑO EXPERIMENTAL.....	93
11.3 RESULTADO DEL DISEÑO EXPERIMENTAL	94
11.4 ANÁLISIS DEL DISEÑO EXPERIMENTAL.....	95
11.5 COMPARACIÓN DEL RENDIMIENTO DEL ALGORITMO.....	100
12. CONCLUSIONES.....	103
13. RECOMENDACIONES	104
BIBLIOGRAFÍA.....	105
ANEXOS	113

LISTA DE TABLAS

	Pág.
Tabla 1. Parámetros iniciales para el desarrollo del problema de TSP.....	71
Tabla 2. Decodificación de la solución codificada.....	72
Tabla 3. Codificación del vector posición obtenido.	72
Tabla 4. Parámetros utilizados para la resolución del problema del TSP.	74
Tabla 5. Costo y tiempo computacional para un número de escenarios 50 y 100 con una cantidad de 100 clientes.....	80
Tabla 6. Costo y tiempo computacional para un número de escenarios 50 y 100 con una cantidad de 200 clientes.....	80
Tabla 7. Factores y niveles.	82
Tabla 8. Instancias.....	83
Tabla 9. Comparación en la inicialización de las espirales.	86
Tabla 10. Combinaciones de prueba	93
Tabla 11. Configuración de los valores de los niveles de cada factor.....	93
Tabla 12. Tratamientos a utilizar en el diseño experimental	94
Tabla 13. Características del equipo de cómputo.	94
Tabla 14. Resultado del diseño experimental.	95
Tabla 15. Efectos estimados para la función objetivo.	96
Tabla 16. Resultados del valor de la función objetivo con Optimización por Espiral e Híbrido de Optimización por Espiral.....	101
Tabla 17. Resultados de la función objetivo con Híbrido de Optimización por espiral y EPSO.....	102

LISTA DE FIGURAS

	Pág.
Figura 1. Problema desde el punto de vista computacional.....	35
Figura 2. Intercambio 3-opt.....	41
Figura 3. Aplicación del procedimiento Or-opt para 3 nodos.	42
Figura 4. Cruce de 1 punto.	48
Figura 5. Cruce de 2 puntos.....	49
Figura 6. Cruce uniforme.	50
Figura 7. Espiral logarítmica	52
Figura 8. Rotación de un vector x.	53
Figura 9. Rotación alrededor del eje z.	54
Figura 10. Ilustración del problema del agente viajero.....	70
Figura 11. Escenarios para el cálculo de la función objetivo.	76
Figura 12. Costos vs número de escenarios para 100 clientes	78
Figura 13. Costos vs número de escenarios para 200 clientes	79
Figura 14. Mutación Flip.	88
Figura 15. Mutación Swap.	88
Figura 16. Mutación Slide.	89
Figura 17. Aplicación de 2-Opt.....	90
Figura 18. Diagrama de Pareto para la instancia 1.....	96
Figura 19. Diagrama de Pareto para la instancia 2.....	97
Figura 20. Diagrama de Pareto para la instancia 3.....	97
Figura 21. Diagrama de Pareto para la instancia 4.....	98
Figura 22. Diagrama de Pareto para la instancia 5.....	98
Figura 23. Diagrama de Pareto para la instancia 6.....	99
Figura 24. Diagrama de Pareto para la instancia 7.....	99
Figura 25. Diagrama de Pareto para la instancia 8.....	100

LISTA DE ANEXOS

	Pág.
ANEXO A. FUNCIONES DE PRUEBA	113

RESUMEN

TÍTULO:

“SOLUCIÓN DEL PROBLEMA DE RUTEO DE VEHÍCULOS CON DEMANDAS ESTOCÁSTICAS (VRPSD) MEDIANTE LA METAHEURÍSTICA OPTIMIZACIÓN POR ESPIRAL”¹

AUTORES:

GELVES TELLO, Natalia Alejandra
MORA MORENO, Ricardo Andrés²

PALABRAS CLAVE:

Ruteo de vehículos; Demandas estocásticas; Metaheurísticas; Optimización por espiral.

DESCRIPCIÓN:

En la presente investigación se implementó la metaheurística denominada Optimización por Espiral para la solución del Problema de Ruteo de Vehículos con Demandas Estocásticas (VRPSD), con el enfoque a priori y la estrategia de reabastecimiento preventivo para un solo vehículo. Esta metaheurística, aprovecha la geometría de las espirales logarítmicas para mejorar las estrategias de búsqueda de intensificación y diversificación. Para el mejoramiento del método, se realizó la inicialización de las rutas mediante la heurística del vecino más cercano y posteriormente se utilizó la mutación, un operador evolutivo y el intercambio 2-Opt, una heurística de búsqueda local, con el fin de mejorar las estrategias de búsqueda de diversificación e intensificación, respectivamente.

Por otra parte, se realizó un diseño de experimentos ²³, con el fin de determinar la influencia de cada factor en la función objetivo. Este análisis se llevó a cabo en 8 instancias diferentes, las cuales fueron diseñadas y desarrolladas en investigaciones realizadas previamente por otros autores. Posteriormente, se verificó el desempeño superior del algoritmo mejorado respecto al algoritmo estándar. Finalmente, se compararon los resultados obtenidos con los arrojados por el algoritmo híbrido EPSO, con el objetivo de probar la eficiencia y eficacia del algoritmo desarrollado. Esta comparación evidenció que el método propuesto obtiene mejores resultados en todas las instancias, con mejoras de hasta el 5,71%.

¹ Trabajo de grado

² Facultad de Ingenierías Físico mecánicas. Escuela de Estudios Industriales y Empresariales. Director: Ph.D. Henry Lamos Díaz.

ABSTRACT

TITLE:

“SOLVING THE VEHICLE ROUTING PROBLEM WITH STOCHASTIC DEMANDS (VRPSD) BY METAHEURISTIC SPIRAL OPTIMIZATION”³

AUTHORS:

GELVES TELLO, Natalia Alejandra
MORA MORENO, Ricardo Andrés⁴

KEY WORDS:

Vehicle routing; Stochastic demands; Metaheuristics; Spiral optimization.

DESCRIPTION:

In the present research work, we implemented the metaheuristic Spiral Optimization for solving the Vehicle Routing Problem with Stochastic Demands (VRPSD), with preventing restocking and employing an a priori strategy. This metaheuristic, uses the geometry of the logarithmic spiral to improve search strategies as intensification and diversification. In order to improve the results, the Nearest Neighbor heuristic was used to create the a priori route. Moreover, we applied mutation (an evolutionary operator) and exchange 2-Opt (a local search heuristic) in order to find better solutions by searching into a wide region and by looking around a good solution in an intensive way, respectively.

On the other hand, it was carried out a design of experiments 2^3 , in order to determine the effect of each input parameter on the objective function. The eight instances used for this DOE, were designed and developed in previous researches by other authors. Later, we verified the higher performance for the improved algorithm over the standard one, by comparing the results obtained with both algorithms. Finally, the final solutions obtained were compared with the ones obtained using the hybrid algorithm EPSO for proving the efficacy and efficiency of the developed method. The comparison showed that the proposed method, obtain better solutions in all instances and improvements of up to 5,71%.

³ Trabajo de Grado.

⁴ Facultad de Ingenierías Físico mecánicas. Escuela de Estudios Industriales y Empresariales. Director: Ph.D. Henry Lamos Díaz.

INTRODUCCIÓN

“La logística de la cadena de suministro es un área de la administración que se ha observado absorbe entre un 60% y un 80% de cada dólar que vende la empresa y que puede ser esencial para su estrategia competitiva y la generación de ingresos”⁵. Este término se atribuye al conjunto de actividades como transporte y control de inventarios que se realizan para obtener la transformación de materia prima a producto terminado y ofrecerle valor agregado a los clientes.

El transporte es una de las actividades más representativa de la cadena de suministro debido a que actualmente las empresas no logran operar sin el movimiento de sus materias primas y/o de sus productos terminados; además de esto, el costo que genera dicha actividad es alrededor del 50% de los costos totales de la cadena de suministro.⁶ Es por esto que es importante que las empresas realicen estudios detallados sobre el transporte de mercancías para lograr la creación de una estrategia de distribución que apunte a la minimización de los costos producidos por transporte y a su vez, disminuya notoriamente los costos atribuidos a la cadena de suministro.

Actualmente, el Gobierno Nacional se encuentra realizando acuerdos comerciales con diversos países como Estados Unidos a fin de aumentar el desarrollo de los sectores logísticos, de infraestructura y transportador⁷. Así mismo, en el Plan Nacional de Desarrollo 2010-2014 se establece como objetivo apoyar el desarrollo del transporte y la infraestructura del país, fortaleciendo las acciones de implantación de la Política Nacional Logística (PNL) con el objeto de aumentar la competitividad del país. Los hechos planteados anteriormente muestran una clara oportunidad para desarrollar mayor investigación en estrategias que garanticen la

⁵ BALLOU, Ronald H. Logística. Administración de la cadena de suministro. 5° Edición. Prentice Hall. Enrique Quintanar Duarte. México, 2004. p xxi. (ISSN 970-26-0540-7)

⁶ Ibid., p. 175.

⁷Revista de logística [en línea]. LEGIS. Mayo-Julio 2012. En: <<http://www.revistadelogistica.com/transporte-de-carga-una-cuanta-aun-a-medio-pagar.asp>> [citado el 28 Julio 2014].

disminución de los costos de transporte debido al aporte que está ofreciendo el gobierno respecto a este tema.

El **problema de ruteo de vehículos (VRP, Vehicle Routing Problem)** es un problema que trata de minimizar los costos de transporte que se presentan en la logística de la cadena de suministro. El VRP consiste en un conjunto de clientes con demandas específicas, un depósito y una flota de vehículos con una capacidad determinada de forma que se pretende encontrar la ruta que minimice costos y que comience y termine en el depósito. En este caso, la capacidad y las demandas son factores determinísticos⁸. A partir del VRP clásico, surgen muchas variantes, entre ellas el **problema de ruteo de vehículos estocásticos (VRPS, Vehicle routing problema stochastic)**, el cual ofrece la posibilidad de establecer parámetros como la demanda o el número de clientes, de forma estocástica.

Dados los distintos factores externos que afectan las decisiones tanto de las empresas como de los clientes, un problema de ruteo de vehículos donde todos los parámetros estén dados por sentado no asegura un correcto acercamiento y por ende, una correcta solución al problema que se está presentado en la vida real. Es así como el **problema de ruteo de vehículos con demandas estocásticas (VRPSD, vehicle routing with stochastic demands)** logra ampliar su campo de acción, aumentando la confiabilidad de la solución ofrecida por el mismo.

Siendo tanto el VRP como cada una de sus variantes, problemas de optimización combinatoria con una complejidad computacional **NP-HARD**⁹, se hace necesario la aplicación de algoritmos que garanticen la obtención de una solución confiable en un tiempo computacional razonable. Es así como han surgido distintas metaheurísticas como La Optimización por Espiral (Spiral Optimization) que

⁸ DAZA, Julio Mario; MONTOYA, Jairo R; NARDUCCI, Francesco. Escuela de Ingeniería de Antioquia, Medellín (Colombia). Resolución del problema de enrutamiento de vehículos con limitaciones de capacidad utilizando un procedimiento metaheurístico de dos fases. En: Revista EIA, No 12, p. 23-38. Diciembre 2009 (ISSN 1794-1237)

⁹ Ibid.

fundamentada en fenómenos naturales como un torbellino o un tornado logra acercarse a una respuesta óptima del problema¹⁰.

La presente investigación se enfocará en el uso de la metaheurística Optimización por Espiral para la solución de VRPSD. Las instancias a utilizar serán las planteadas en el trabajo de investigación “Optimización por simulación basado en un sistema evolutivo de optimización de enjambre de partículas para el problema de ruteo de vehículos con demandas estocásticas” realizado por la ingeniera Silvia Adriana Galván Núñez¹¹, con el fin de confrontar los resultados alcanzados y verificar la efectividad del algoritmo.

¹⁰ TAMURA, Kenichi; YASUDA, Keiichiro. Spiral Optimization. A new multipoint search method. En: Systems, Man, and Cybernetics (SMC). Octubre 2011. 6 p.

¹¹GALVÁN NÚÑEZ, Silvia Adriana; ARIAS, Javier; LAMOS, Henry. Optimización por simulación basado en un sistema evolutivo de optimización de enjambre de partículas para el problema de ruteo de vehículos con demandas estocásticas. Tesis de maestría de ingeniería industrial. Bucaramanga. Universidad Industrial de Santander. 2012.

1. PLANTEAMIENTO DEL PROBLEMA

El problema de ruteo de vehículos (Vehicle Routing Problem, VRP) tiene como objetivo determinar la ruta óptima para un conjunto de vehículos que inician y terminan su recorrido en un depósito, el cual proporciona abastecimiento a los mismos teniendo en cuenta que los vehículos cuentan con una capacidad fija. Se define como ruta óptima a aquella que minimiza los costos de transporte y satisface la demanda de los clientes, los cuales estarán ubicados en diferentes puntos geográficos.

El VRP no captura un aspecto importante de los problemas de distribución de la vida real, debido a que varios de los parámetros del problema (demanda, ubicaciones de los clientes, entre otros) no se conocen con certeza por lo que no se deberían definir como determinísticos. Esto da lugar al ruteo de vehículos estocásticos (Stochastic Vehicle Routing Problems, SVRP)¹², el cual proporciona la libertad de definir a uno o varios parámetros como estocásticos.

El presente proyecto basará sus estudios en el problema de ruteo de vehículos con demandas estocásticas (Vehicle Routing Problem with Stochastic Demands, VRPSD), el cual es una subdivisión del SVRP y se caracteriza porque la única variable estocástica es la demanda de los clientes. Así mismo, ésta sigue una distribución discreta y sólo se conoce cuando el vehículo ha llegado a la ubicación.

Como consecuencia de la capacidad fija de los vehículos, en algún lugar la carga de estos no será suficiente para satisfacer la demanda exigida por el siguiente cliente por lo cual el vehículo deberá regresar al depósito para realizar un **reabastecimiento preventivo**. Así mismo, es posible que se realice un **abastecimiento por fallo en la ruta**, lo que sugiere que no se cuenta con la cantidad que el cliente exige, haciendo necesaria la ida al depósito para volver al

¹² FAN, Jianhua; et al.; Op. Cit., p.1688

cliente no satisfecho y entregar la cantidad demandada a fin de continuar con la ruta.¹³

Para instancias de gran tamaño, se requiere la aplicación de métodos aproximados a fin de obtener una solución cercana a la óptima. La razón de esto es que el VRPSD tiene una complejidad computacional **NP-HARD**¹⁴ lo cual representa: Tiempos de procesamiento largos, espacio computacional considerable y la inexistencia de un algoritmo polinomial que ofrezca una solución óptima a dicho problema.

Es así como en la literatura existen diversas investigaciones en las cuales se han aplicado múltiples metaheurísticas con el fin de solucionar el VRPSD. Por este motivo, a fin de comparar los resultados obtenidos y definir las distintas instancias para el presente proyecto, se tomará como base el artículo “Optimización por simulación basado en epso para el problema de ruteo de vehículos con demandas estocásticas”¹⁵.

Finalmente, el desarrollo de nuevos métodos evolutivos, han aumentado el interés por aplicarlos en distintos problemas de optimización combinatoria como el VRP a fin de verificar la efectividad de los mismos. De esta manera, para solucionar el VRPSD, se aplicará la metaheurística optimización por espiral, basada en la geometría de las espirales logarítmicas, propias de fenómenos naturales como lo son los tornados.¹⁶

¹³ GALVÁN NÚÑEZ; Op. Cit.

¹⁴ IGLESIAS MARTÍNEZ, José; y otros. Complejidad Computacional. Grado Ingeniería informática. Teoría de Autómatas y Lenguas formales. [En línea] <<http://ocw.uc3m.es/ingenieria-informatica/teoria-de-automatas-y-lenguajes-formales/material-de-clase-1/tema-8-complejidad-computacional>> En: Universidad Carlos III de Madrid. [Citado el 12 de Julio del 2014].

¹⁵ GALVÁN NÚÑEZ; Op. Cit.

¹⁶ TAMURA; Op.Cit., 6p.

2. JUSTIFICACIÓN DEL PROBLEMA

El Ruteo de Vehículos con Demandas Estocásticas (VRPSD) ha logrado responder a problemas de logística de distribución que presentan una complejidad mayor a los problemas que resuelve el VRP debido a que permite la existencia de una incertidumbre en la demanda de los clientes a satisfacer; de forma tal que amplía su campo de acción dado que logra resolver problemas más cercanos a la realidad, donde no todos sus parámetros son determinísticos o trabajados con absoluta certeza, por ejemplo, la demanda de los clientes al presentar variabilidad en el día a día de las empresas, es usual modelarla como una variable aleatoria.

A partir de lo planteado anteriormente, se ha generado un gran interés alrededor a esta problemática, reflejado en las investigaciones realizadas a lo largo de la historia dentro de las cuales se pueden destacar los estudios de Tillman (1969) quien propone una metodología de solución mediante el algoritmo Clarke and Wright's savings, Cheong et al. (2006) el cual presentó un algoritmo evolutivo que incorpora una heurística específica para la explotación local y un método de simulación de ruta para evaluar la idoneidad de la solución para VRPSD y Dong y Xiang (2006) quienes propusieron un nuevo algoritmo híbrido que combina la metaheurística inspirada en las hormigas con la heurística de optimización local¹⁷.

A partir de la investigación reciente de la ingeniera Silvia Adriana Galván Núñez, como tesis de maestría en ingeniería industrial de la UIS y los resultados obtenidos, es de interés para el grupo de investigación OPALO, la aplicación de una nueva metaheurística denominada "Optimización por espiral" la cual utiliza la geometría de las espirales logarítmicas, propias de fenómenos naturales como lo son los tornados y huracanes, para ampliar su rango de búsqueda en la fase inicial

¹⁷ TRIPATHI, Op. Cit., p. 2476-2487.

(diversificación) e intensificar la búsqueda alrededor de una buena solución en la fase final (intensificación)¹⁸.

¹⁸ TAMURA, Op.Cit., 6 p.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Aplicar la metaheurística Optimización por Espiral para resolver el problema de Ruteo de Vehículos de Demandas Estocásticas (VRPSD).

3.2 OBJETIVOS ESPECÍFICOS

- Realizar una revisión de la literatura del problema de ruteo de vehículos con demandas estocásticas (VRPSD).
- Desarrollar un algoritmo de Optimización por espiral utilizando el software MATLAB.
- Confrontar los resultados alcanzados mediante la metaheurística de Optimización por espiral con las soluciones obtenidas por métodos previamente realizados por otros investigadores.
- Elaborar un artículo publicable en base al trabajo de investigación realizado.

4. REVISIÓN DE LA LITERATURA

Formalmente, los inicios del problema de ruteo de vehículos se remontan a **1959** cuando **Dantzing y Ramser**¹⁹ realizaron una generalización del problema del agente viajero (*Travelling Salesman Problem*, TSP) y publicaron un artículo en el cual modelaban el despacho de gasolina a varias estaciones de servicio por medio de una flota de camiones, desde una terminal.

Posteriormente **Stewart, William y Golden, Bruce**²⁰ en **1983**, estudiaron y modelaron el problema de ruteo de vehículos estocástico (*Stochastic Vehicle Routing Problem*, SVRP). Además, presentaron dos enfoques para resolver los modelos SVRP: el primero se basa en transformar el SVRP en un VRP equivalente y aplicar las heurísticas existentes para el mismo. El segundo, consiste en la adaptación de las heurísticas existentes al SVRP. Para este último, se realiza la presentación y adaptación dentro del artículo de los algoritmos Clarke and Wright y GLM.

Dror, Moshe; Laporte, Gilbert y Trudeau, Pierre²¹ en **1989** estudiaron una clase del SVRP en la cual solamente es incierta la demanda de los clientes (*Vehicle Routing Problem with Stochastic Demand*, VRPSD). En este artículo, se describen las características, los parámetros, las políticas de operación y servicio de este problema. Igualmente, los autores proponen dos nuevos sistemas de solución para el mismo: el primero corresponde a un modelo de programación entera y el segundo es un proceso de decisión markoviano.

¹⁹ ROCHA, L.; GONZÁLEZ, C. y ORJUELA, J. Una revisión al estado del arte del problema de ruteo de vehículos: Evolución histórica y métodos de solución. En: Ingeniería. 2011. Vol. 16, No. 2, p. 35 - 55.

²⁰ STEWART, W y GOLDEN, B. Stochastic vehicle routing: A comprehensive approach. En: European Journal of Operational Research. Diciembre, 1983. Vol. 14, p. 371-385.

²¹ DROR, M.; LAPORTE, G. y TRUDEAU, P. Vehicle routing with stochastic demands: Properties and solution frameworks. En: Transportation Science. Agosto 1989. Vol. 23, No. 3, p. 166.

Secomandi, Nicola²² en **1998** considera el problema de ruteo de vehículos con demandas estocásticas y estudia la aplicabilidad de la programación neuro-dinámica (*Neuro-Dynamic Programming, NDP*) en la resolución de problemas de optimización combinatoria estocástica. Así mismo, realiza la comparación de dos algoritmos NDP: iteración de política aproximada optimista (*Optimistic Approximate Policy Iteration, OAPI*) y política de despliegue (*Rollout Policies, RP*). Dentro de los resultados obtenidos, se destaca el mejoramiento de una política de vecino más cercano en un 2,3% mediante la OAPI. A partir de esto, se concluye que la RP produce soluciones con una mayor calidad.

Yang, Wen-Huei; Mathur, Kamlesh y Ballou, Ronald²³ en el **2000** incorporan una política óptima de reabastecimiento de existencias del vehículo en el diseño de la ruta. El objetivo de la misma es minimizar el costo total esperado de la ruta al ubicar deliberadamente puntos de abastecimiento con los cuales la probabilidad de fallo de ruta y el costo de reabastecimiento sean reducidos. Así mismo, desarrollaron y comprobaron la eficiencia y robustez de dos algoritmos heurísticos: *route-first-cluster-next* y *cluster-first-route-next*.

Secomandi, Nicola²⁴ presenta en el **2002** un algoritmo del enfoque de despliegue para problemas de secuenciación estática en base a una heurística cíclica empleada por Haimovich y Rinnooy Kan (1985) y por Bertsimas (1992) en la investigación del problema del enrutamiento de vehículos. Posteriormente, extiende esta aplicación a problemas inmersos en ambientes dinámicos y estocásticos por lo que examina la aplicabilidad de la misma al problema de ruteo de vehículos con demandas estocásticas. Finalmente, se encuentra que los resultados obtenidos por el enfoque de despliegue superan a los de la heurística cíclica.

²² SECOMANDI, Nicola. Comparing neuro-dynamic programming algorithms for the vehicle routing problem with stochastic demands. En: Computers & Operations Research. 2000. Vol. 27, p. 1203.

²³ YANG, Wen-Huei; MATHUR, Kamlesh; BALLOU, Ronald. Stochastic vehicle routing problem with restocking. En: Transportation Science Trans. Sci. 2000. Vol. 34, No.1, p. 99-112.

²⁴ SECOMANDI, Nicola. Analysis of a Rollout Approach to Sequencing Problems with Stochastic Routing Applications. En: Journal of Heuristics. Septiembre 2003. Vol. 9, No. 4, p. 321-352.

Chepuri, Krishna; Homem-De-Mello, Tito²⁵ en el **2005** desarrollaron una nueva heurística a fin de solucionar el problema del VRPSD, usando el método de entropía-cruzada (*Cross-Entropy*, CE). Además, se incluye el método Monte Carlo a fin de manejar la incertidumbre inherente de los problemas de optimización estocástica, obteniendo mejores estimaciones de la función objetivo. Finalmente, se evidencia una de las grandes ventajas del algoritmo CE y es la efectividad del mismo para solucionar múltiples problemas de ruteo de vehículos.

Bianchi, Leonora; Birattari, Mauro; Chiarandini, Marco; Manfrin, Max; Mastrolilli, Monaldo; Paquete, Luis; Rossi-Doria, Olivia y Schiavinotto, Tommaso²⁶ en el **2006** encontraron que rápidas aproximaciones a la función objetivo podían proporcionar mayor tiempo para la búsqueda de soluciones de buena calidad. Para esto se realizó la hibridación del proceso de búsqueda de la metaheurística al intercambiar la función objetivo exacta del problema del VRPSD por dos aproximaciones (*VRPSD approximation* y *TSP approximation*). Las metaheurísticas seleccionadas para este estudio fueron: el recocido simulado, la búsqueda tabú, la búsqueda local iterada, la optimización de colonia de hormigas y los algoritmos evolutivos. Así mismo, los autores investigaron dos tipos de hibridación: la primera empleaba un algoritmo de búsqueda local (Or-Opt) y se aplicaba tanto para la aproximación TSP como para la VRPSD. La segunda empleaba la analogía TSP (*TSP analogy*) y el uso del operador de búsqueda local 3-opt, con grandes resultados solamente para el TSP.²⁷

Fan, Jianhua; Wang, Xiufeng²⁸ presentaron en el **2006** una heurística para la solución del VRPSD con múltiples vehículos. La metodología de solución de la misma se dividió en dos fases: en la primera, se propone una heurística *cluster*

²⁵ CHEPURI, Krishna;DE-MELLO, Tito. Solving the Vehicle Routing Problem with Stochastic Demands using the Cross-Entropy Method.En: Annals of Operations Research.Febrero 2006. Vol.134, No. 1, p. .153-181.

²⁶ BIANCHI, Leonora, et al. Hybrid Metaheuristics for the Vehicle Routing Problem with Stochastic Demands.En: Journal of Mathematical Modelling and Algorithms. 2006. P. 91-110.

²⁷ Ibid.

²⁸ FAN, Jianhua; et al. A Multiple Vehicles Routing Problem Algorithm with Stochastic Demand.En: 6th World Congress on Intelligent Control and Automation (2006, Dalian), p. 1688-1692.

first-route next y en la segunda, se refina la solución obtenida por esta heurística usando la política de reoptimización llamada algoritmo *rollout*. Finalmente, en base a los resultados obtenidos, se evidencia la efectividad del algoritmo para problemas con demanda estocástica.

Tan; K.C.; Cheong, C.Y. y Goh, C.K.²⁹ en el **2007** solucionaron el problema del VRPSD con limitaciones de capacidad y tiempo. De esta forma, optimizaron un problema multiobjetivo, minimizando la distancia recorrida, la remuneración del conductor y el número de vehículos requeridos. Para esto, los autores emplearon un algoritmo evolutivo multiobjetivo (MOEA) con operadores genéticos mejorados, el esquema de clasificación de Pareto, y dos heurísticas de búsqueda local: *Shortest Path Search* (SPS) y *Which Directional Search* (WDS). Adicionalmente se propone un método de simulación de ruta (RSM) para determinar el costo estocástico de una solución VRPSD.

Ak, Aykagan; Erera, Alan L³⁰ presentan en el **2007** la aplicación del método heurístico búsqueda Tabú para la solución del problema de ruteo de vehículos con demandas estocásticas. La importancia del aporte y la investigación presentada en este artículo parte del cambio de estrategia utilizada en cuanto a los vehículos; la mayoría de estudios realizados sobre este problema plantean que los vehículos trabajan de forma independiente. En este caso, los investigadores consideran que los vehículos trabajan en pares, estrategia que denominan vehículos emparejados para el ruteo de vehículos con demandas estocásticas. También cabe resaltar que dada la estrategia utilizada en este estudio, se establece que la distribución que sigue la demanda de los clientes es una distribución homogénea.

²⁹ TAN, K; CHEONG, C; GOH, C. Solving multiobjective vehicle routing problem with stochastic demand via evolutionary computation. En: European Journal of Operational Research. Marzo 2007. Vol.177, No. 2, p. 813-839.

³⁰ AK, Aykagan; ERERA, Alan L. A Paired- Vehicle Recourse Strategy for the Vehicle-Routing Problem with Stochastic Demands. En: Transportation science. Vol. 41, Mayo 2007, p. 222-237.

Dedong, Wang; Qijun, Chen; Lili³¹ en el **2008** realizan un estudio basado en el problema de ruteo de vehículos estocásticos. En este caso se plantea que la demanda de los clientes es estocástica y además de eso, el número de clientes que se deben atender también es una variable aleatoria. Este problema es denominado VRPSCD (Vehicle Routing Problem with Stochastic Customers and Demands) y especifica que sólo se sabrá si existe el siguiente cliente justo en el momento en el que se atiende al cliente del nodo precedente; así mismo, la demanda del cliente sólo se sabe en el momento en que se va a atender. Para la solución de este problema, los investigadores plantean un método de búsqueda aleatoria que garantiza una convergencia rápida y evita la idea de quedar atrapado en los mínimos locales; esta búsqueda se basa en el algoritmo de recocido simulado el cual ofrece una respuesta bastante cercana a la óptima y el algoritmo genético quien es el que garantiza la rapidez de la convergencia y ofrece una fuerte búsqueda global; es así como surge el nuevo algoritmo genético de recocido simulado para la solución del VRPSCD.

Peng, Yong; Zhu, Hai-ying³² ofrecen en el **2008** un algoritmo de optimización por enjambre de partículas con una programación dinámica PSO-DP para la solución del problema de ruteo de vehículos con demandas estocásticas (VRPSD). Mediante el algoritmo PSO se incorpora un operador denominado Inver-over, el cual ayuda a optimizar la ruta a priori debido a que la función de éste es seleccionar la solución tomando como guía la población actual, es decir que antes de seleccionar la solución óptima se compara con la población hasta garantizar que es ésta la mejor³³; con la programación dinámica se logra calcular el valor de la función objetivo, es decir, el costo esperado de la solución.

³¹ DEDONG, Wang; QIJUM, Chen; Lili. Stochastic Vehicle Routing Problems and their Solution Algorithm. En: International Conference on Automation and Logistics. Qingdao, China. Septiembre, 2008, p. 958-962.

³² PENG, Yong; SHU, Hai-ying. Research on Vehicle Routing Problem with Stochastic Demand and PSO-DP Algorithm with Inver-over Operator. En: ScienceDirect. Systems Engineering- theory & Practice. Vol. 28. Octubre 2008, p. 76-81.

³³ MINETTI, Gabriela F. Una solución de computación evolutiva para el TSP, su posible aplicación en las organizaciones. Capítulo 7: un algoritmo evolutivo avanzado para TSP. En: Universidad Nacional de la Plata. Facultad de Informática. Magister en automatización de Oficinas. Octubre de 2000, p. 84-92.

Novoa, Clara; Storer, Robert³⁴ estudian en el **2008** los distintos algoritmos de programación dinámica aproximados para el VRPSD desde una perspectiva dinámica y de re-optimización. Mediante la simulación Monte Carlo logran determinar el “costo de ir” y examinar lo que sucede cuando hay una falla en la ruta; En este caso, el vehículo que presenta falla en la ruta puede completar la demanda del cliente en otra visita que puede no ser inmediata, lo que disminuye el costo de enrutamiento de los vehículos y genera una re-optimización del problema.

En este estudio se logra determinar que con la aplicación del algoritmo de despliegue en dos etapas se reduce el costo en aproximadamente 4,8% y se disminuye el tiempo de cálculo en un 65%; el desarrollo de las mejores secuencias de enrutamiento y el desarrollo de un sistema de poda genera una disminución en los costos de un intervalo de 3,56%-4,11% comparado con soluciones anteriores del problema.

Tripathi, Mukul; Kuriger, Glenn; Wan, Hung-da³⁵ presentan en el **2009** una nueva metaheurística basada en el algoritmo de optimización mediante colonia de hormigas (ACO); este nuevo algoritmo denominado Neighborhood Search embedded Adaptive Ant Algorithm (ns-AAA) muestra una mejora en la calidad de la solución así como una considerable disminución del tiempo de duración de los cálculos. Para la verificación del funcionamiento de dicho algoritmo, se hicieron pruebas sobre unas bases de datos obtenida por un diseño de experimentos (DOE) de forma tal que se pudo comparar los resultados obtenidos con éste y los obtenidos por el método ACO y el algoritmo genético (GA).

³⁴ NOVOA, Clara; STORER, Robert. An approximate dynamic programming approach for the vehicle routing problem with stochastic demands. En: European Journal of Operational Research. 2008, p. 509-512.

³⁵ TRIPATHI, Mukul; KURIGER, Glenn; WAN, Hung-da. An ant based simulation optimization for vehicle routing problema with stocastic demands. En: Winter Simulation Conference. University of Texas at San Antonio. 2009, p. 2476-2487.

Ismail, Zuhaimy; Irhamah³⁶ proponen en el **2010** un nuevo algoritmo híbrido HGATS entre el algoritmo genético (AG) y la búsqueda Tabú (TS) para la solución de un problema de recolección de residuos sólidos. Siendo éste un problema de ruteo de vehículos con demandas estocásticas, se aplica el híbrido HGATS junto con la aplicación del GA y TS dando como resultado que el HGATS logra ser superior a la búsqueda Tabú y similar al algoritmo genético cuando el tamaño de la población es pequeño mientras que cuando la población es grande, HGATS logra ser superior al algoritmo genético y similar al TS.

Sulieiman, Dalia; Jourdan, Laetitia; Talbi, El-Ghazali³⁷ en el **2010**, dada la necesidad de tener soluciones robustas para los problemas de VRPSD, plantean dos modelos biobjetivo que serían resueltos por tres algoritmos multiobjetivo con el fin de determinar cuál de los dos modelos satisface de mejor manera el problema y así mismo, verificar cuál de los tres algoritmos presenta una solución más robusta en cada uno de los modelos.

El primer modelo biobjetivo es el modelo de entropía, el cual tiene como principal objetivo minimizar la distancia esperada que se desplaza el vehículo y como segundo objetivo minimizar la entropía de la distancia; el segundo modelo es el relacionado con la desviación estándar, el cual tiene como primer objetivo minimizar la distancia recorrida por los vehículos y como segundo objetivo tiene reducir al mínimo la desviación de la distancia media de las vías. En cuanto a los algoritmos, el primero a utilizar es el indicador basado en el algoritmo evolutivo (IBEA), el segundo algoritmo es el conocido NSGAII y por último, se aplicaría el algoritmo multiobjetivo genético MOGA.

Para determinar la robustez de la solución, se debía encontrar la diferencia entre la distancia máxima recorrida (peor caso) y la distancia mínima recorrida (mejor

³⁶ ISMAIL, Zuhaimy; IRHAMAH. Genetic Algorithm and Tabu Search for Vehicle Routing Problems with Stochastic Demand. En: International Conference on Mathematical Science. Instituto Americano de Física. 2010, p. 488 – 504.

³⁷ SULIEMAN, Dalia; JOURDAN, Laetitia; TALBI, El-Ghazali. Using multiobjective metaheuristics to solve VRP with uncertain demands. En: Evolutionary Computation (CEC). 2010, 8 p.

caso) para cada ruta ofrecida por los algoritmos. Es así como se concluye que el modelo de la entropía da soluciones más robustas que el modelo sigma o de la desviación estándar; En cuanto a los algoritmos, con el modelo de entropía no había una diferencia marcada entre los tres mientras que con el modelo sigma, el algoritmo IBEA daba respuestas más robustas que el MOGA y el NSGAI.

S, Geetha; G, Poonthalir; P. T, Vanathi³⁸ en el año **2010** presentan una solución para el problema VRP basados en una metaheurística PSO híbrida, añadiéndole a ésta la codificación de las partículas. La importancia de este artículo se basa en la forma como codifican y decodifican cada una de las partículas; dado que el PSO comienza con la inicialización de las mismas, es decir asignándoles una posición aleatoria a cada una de ellas, los autores aplican un método para garantizar que esa asignación sea de mayor calidad y con mayor diversificación. Es así como establecen una permutación de rutas que posteriormente transforman en posiciones de partículas. A partir de esto, aplican el algoritmo perteneciente a la metaheurística PSO y al final decodifican las nuevas posiciones de las partículas para obtener la ruta que debe realizar el vehículo.

JC, Goodson; JW, Ohlmann; BW, Thomas³⁹ en el año **2012** examinaron estructuras vecinales o vecindarios para la solución de problemas de ruteo de vehículos (VRP) mediante una codificación cíclica. Esta codificación lo que hace es una permutación de los clientes para encontrar un conjunto de soluciones para el VRP; posterior a esto, mediante un algoritmo de tiempo polinomial logra encontrar la mejor solución al VRP. Para el problema de ruteo de vehículos con demandas estocásticas (VRPSD), se añade un procedimiento de recocido simulado que mejora la solución del VRP clásico ofrecido en la primera parte y se agregan los costos por fallos de ruta.

³⁸ GEETHA, S.; POONTHALIR, G.; VANATHI, P.T. Metaheuristic approach for the Multi-depot Vehicle Routing Problem. En: Applied Artificial Intelligence. Vol. 26, No. 9, 2012, p. 878-901.

³⁹ GOODSON, Justin C.; OHLMANN, Jeffrey W.; THOMAS, Barrett W. Cyclic-order neighborhoods with application to the vehicle routing problem with stochastic demand. En: European Journal of Operational Research, vol. 217, 2012, p. 312–323.

Marinakis, Tannis; Marinaki, Magdalene⁴⁰ en el **2013** basan su estudio en la modificación del algoritmo Optimización por enjambre de partículas (PSO) para la aplicación del mismo al problema de ruteo de vehículos con demandas estocásticas (VRPSD). De esta investigación surge el nuevo algoritmo The Combinatorial Expanding Neighborhood Topology Particle Swarm Optimization (CENTPSO) el cual logra, en primera instancia la aplicación del PSO a problemas con variables discretas: esto básicamente se logró por la ecuación de posiciones en la cual utilizan la estrategia de volver a vincular el camino y varias velocidades. En segunda instancia, el algoritmo incluye características de la metaheurística búsqueda local lo que genera que su búsqueda comience en un vecindario pequeño y a medida que realiza las corridas añade al vecindario nuevos miembros al punto de incluir todo el enjambre; con esto garantiza una capacidad de exploración local que termina ofreciendo una exploración global del enjambre.

Jabali, Ola; Rei, Walter; Grandeau, Michel; Laporte, Gilbert⁴¹ encaminan en el **2014** su estudio en el problema de ruteo de vehículos con demandas estocásticas (VRPSD). Basados principalmente en los fallos de ruta que se pueden presentar en el problema, dividen el modelo de programación estocástica en dos etapas (la generación de rutas parciales generales y la identificación y separación de las rutas que presentan fallas); posterior a esto, resuelven el problema por medio de un algoritmo entero denominado L-Shaped. Es así como logran una disminución notoria en el número de soluciones factibles y reducen el tiempo de computación llegando a mejores resultados.

⁴⁰ MARINAKIS, Yannis; MARINAKI, Magdalene. Combinatorial Expanding Neighborhood Topology Particle Swarm optimization for the Vehicle Routing Problem with Stochastic Demands. En: GECCO'13. 6 de Julio de 2013, p. 49-56.

⁴¹ JABALI, Ola; REI, Walter; GENDREAU, Michel; LAPORTE, Gilbert. Partial-route inequalities for the multi-vehicle routing problema with stochastic demands. En: ScienceDirect. Discrets Ap.plied Mathematics. 2014, 16 p.

5. MARCO CONCEPTUAL

5.1 OPTIMIZACIÓN COMBINATORIA

La combinatoria, en base a la definición planteada por Lawler (1996), es la rama de la matemática que trata de ordenar objetos usualmente finitos en número y sujetos a varias restricciones⁴². Por otra parte, un problema de optimización busca la maximización o minimización de la función objetivo dentro de la región de soluciones factibles.

Ahora, si el espacio de solución junto con el conjunto de soluciones factibles del problema son discretos, es decir, si son conjuntos de un número finito de elementos, se denomina problema de optimización combinatoria⁴³.

Adicionalmente, el conjunto finito de soluciones que se encuentra en el espacio de búsqueda es denominado S^{44} . El dominio de dicho espacio, afectará el problema de forma que si éste es binario, las variables de decisión serán variables binarias y el problema de optimización también será binario. Es así como:

- Dominio de espacio de búsqueda binario → Problema de Optimización binario.
- Dominio de espacio de búsqueda entero → Problema de Optimización entera.
- Dominio de espacio de búsqueda continuo (los valores de las variables son números reales) → Problema de optimización continua.
- Dominio de espacio de búsqueda mixto (binario, entero y continuo) → Problema de Optimización mixto.

⁴² PASTOR MORENO, Rafael. Metalgoritmo de optimización combinatoria mediante la exploración de grafos. Junio 1999. Tesis doctoral (Ingeniería Industrial). Universitat Politècnica de Catalunya, p. 15.

⁴³ Ibid.

⁴⁴ MARTÍ, Rafael. Procedimientos metaheurísticos en optimización combinatoria. En: Departament d'Estadística i Investigació Operativa. Facultat de Matemàtiques. Universitat de València. 60 p.

A partir de esto, se establecen las siguientes condiciones para denominar a un problema, problema de optimización combinatoria.

- Existencia de una variables de decisión $X = [x_1, x_2, \dots, x_n]$
- Se debe determinar el dominio de cada una de las variables $D_1, D_2, D_3, \dots, D_n$ y así, poder determinar el tipo de optimización combinatoria.
- Deben existir las restricciones entre variables que ayudan a discriminar las soluciones existentes.
- Debe existir una función objetivo f tal que el objetivo del problema sea minimizar o maximizar dicha función⁴⁵.

La solución óptima que se obtiene de este problema debe ser $s_1 \in S$ tal que $f(s_1) \leq f(s_n)$ si se desea minimizar la función objetivo, si se desea maximizar la función objetivo entonces $f(s_1) > f(s_n)$.

Los problemas de optimización combinatoria son muy comunes en la vida real, se presentan tanto en distribución como en la organización de la producción, en el diseño de redes de telecomunicación y en problemas de ingeniería y reingeniería de software⁴⁶. Estos tipos de problemas suelen clasificarse según la dificultad computacional que presentan.

5.2 COMPLEJIDAD COMPUTACIONAL

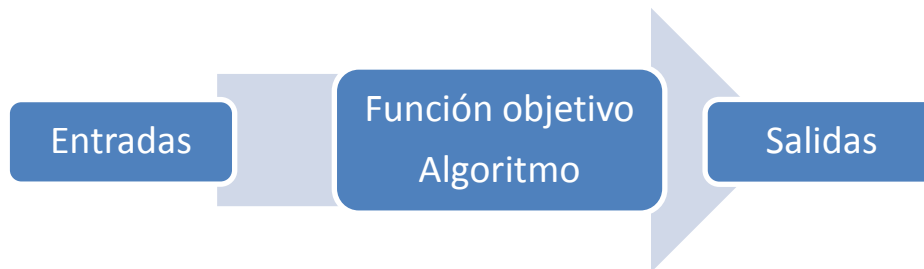
Un problema desde el punto de vista computacional está conformado por unas entradas o parámetros, unas salidas o la solución del problema y la función objetivo perteneciente al problema. Se define problema desde el punto de vista computacional a aquel que se resuelve mediante la relación entre un conjunto de

⁴⁵ BLUM, Christian, y ROLI, Andrea. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. En: ACM Computing Surveys (CSUR), 2008., vol. 35. no. 3., p. 268.

⁴⁶ MARTÍ, Op. Cit., 60 p.

instancias, es decir cada conjunto de posibles combinaciones de los parámetros establecidos en el problema, y un conjunto de soluciones.

Figura 1. Problema desde el punto de vista computacional.



Para determinar las salidas del problema, se debe contar con un algoritmo que logre evaluar cada una de las entradas en la función objetivo y determinar cuales es la mejor solución al problema.

La complejidad computacional determina la eficiencia del algoritmo de acuerdo a la solución del problema. Esta eficiencia suele evaluarse mediante el tiempo que tarda el algoritmo en encontrar la solución o en ofrecer una salida y/o mediante el espacio computacional utilizado por el mismo, donde el espacio es básicamente el almacenamiento de datos en la computadora. De acuerdo a la complejidad computacional, los problemas pueden clasificarse en problemas tratables o problemas no tratables⁴⁷.

⁴⁷ IGLESIAS MARTÍNEZ; Op. Cit.

Un problema es denominado no tratable o intratable⁴⁸ cuando no puede ser resuelto por ningún algoritmo. Esto se da por diferentes motivos entre los cuales se consideran los siguientes:

- El problema es indecidible, es decir, que no existe un criterio de parada que logre dar por terminada la corrida y ofrecen una salida.
- El problema es decidible pero no se conoce ningún algoritmo polinomial que lo resuelva.

Un problema es denominado tratable⁴⁹ cuando se puede resolver en un tiempo polinómico, es decir que el tiempo de duración aumenta en forma polinomial a medida que aumenta el tamaño de los datos de entrada. Estos problemas son denominados problemas **P**.

Existen otros tipos de problemas que, aunque no son problemas **P**, su solución se puede comprobar en un tiempo polinomial; esto quiere decir que estos problemas no se pueden solucionar en este tiempo pero el conjunto de posibles soluciones se pueden comprobar y determinar si son soluciones correctas o no lo son. Un problema de este tipo es denominado problema **NP (Non Deterministic Polynomial time)**. Los problemas NP pueden dividirse en problemas **NP-COMplete** y **NP-HARD**⁵⁰.

El problema **NP-COMplete** tiene como características que es un problema **NP** y que todos los demás problemas de **NP** se pueden reducir a problema **NP-COMplete** añadiéndole un costo en el tiempo polinomial. Es importante resaltar que si se demuestra que existe un problema **NP-COMplete** que pertenece a los

⁴⁸ Ibid.

⁴⁹ Ibid.

⁵⁰ Universidad de Buenos Aires. Complejidad – problemas NP-COMplete. [En línea] <http://www-2.dc.uba.ar/materias/aed3/2011-01/Documents/algo3_compl.pdf> [Citado el 12 de Julio del 2014] En: Universidad de Buenos Aires. Algoritmos y estructuras de datos III.

problemas **P** entonces todos los problemas **NP-COMLETE** pertenecen a los problemas **P**⁵¹.

El problema **NP-HARD** no tiene ningún algoritmo polinómico que lo resuelva, es así como se utilizan algoritmos que ofrezcan una respuesta aproximada a la óptima.

En cuanto a la optimización combinatoria, algunos de estos problemas son fáciles de resolver porque son problemas lineales y por tanto se solucionan mediante métodos exactos; pero la mayoría de los problemas de optimización combinatoria son de difícil solución, considerándose problemas **NP-HARD** por lo que se hace necesaria la aplicación de métodos aproximados.

5.3 MÉTODOS DE SOLUCIÓN

Dada la existencia de los problemas con una complejidad computacional **NP**, surgen diferentes métodos de solución entre los cuales se encuentran los métodos exactos y los métodos aproximados; estos últimos son utilizados para los problemas **NP-HARD** debido a que con las características de estos problemas, el tiempo computacional utilizado para obtener la solución óptima es muy elevado, pero gracias a los métodos aproximados, se logra establecer una respuesta muy cercana a la óptima en un tiempo razonable. En este grupo de métodos se encuentran las heurísticas y las metaheurísticas.

⁵¹ IGLESIAS MARTÍNEZ; Op. Cit.

5.3.1 Métodos Exactos. Los métodos exactos son aquellos capaces de ofrecer la respuesta óptima de un problema determinado en un tiempo determinado. Para problemas con una complejidad computacional NP-HARD suelen tener tiempos de desarrollo bastante elevados (este tiempo crece exponencial al tamaño de la entrada del problema). Algunos de los métodos exactos son nombrados a continuación.

- **Método simplex.** Es un procedimiento desarrollado por George Dantzing en 1947, con gran utilización debido a su extraordinaria eficiencia para solucionar problemas de programación lineal. Este método parte de una solución inicial (si es posible se selecciona el origen), y examina si alguna de las aristas de la posición actual conduce a una tasa positiva de mejoramiento de la función objetivo. En caso de presentarse una mejor solución, se realiza una iteración para moverse a esta posición y se examina la tasa de mejoramiento de cada arista de la misma. El algoritmo finaliza, cuando ninguna de las aristas de la actual posición conlleva a una tasa de mejoramiento positiva, estableciendo esta como la solución óptima⁵².
- **Branch and Bound.** Es uno de los métodos exactos más utilizados para la resolución de problemas de optimización combinatoria **NP-HARD**. Este método funciona a través de un árbol de búsqueda que comienza con un nodo denominado nodo raíz y es quien está directamente relacionado con el problema a resolver. A partir de éste, surgen nuevos nodos correspondientes a sub-problemas que se van optimizando de forma individual y se van ramificando hasta llegar a la solución óptima del problema en cuestión⁵³.

⁵² HILLIER, Frederick S.; LIEBERMAN, Gerald J. Introducción a la investigación de operaciones. 9° Ed. Ciudad de México, México: Mc Graw Hill. 2010, p. 81-88.

⁵³ LI; Hong-Gui; LI, Xing-Guo. Image segmentation with pseudo branch and bound algorithm. En: Proceedings of the Eighth International Conference on Machine Learning and Cybernetics, Baoding, 12-15 Julio de 2009, 5 p.

5.3.2 Métodos Aproximados. Dados los grandes tiempos de procesamiento que ofrecen los métodos exactos a la hora de resolver problemas *NP-HARD*, surgen los métodos aproximados, los cuales ofrecen resultados en tiempos razonables. Aunque estas soluciones no son las óptimas, muchas se consideran soluciones muy cercanas a las óptimas. Los métodos aproximados se clasifican en métodos heurísticos y métodos metaheurísticos.

5.3.2.1 Métodos Heurísticos. Basados en un algoritmo planteado por *Winker and Maringer (2007)*, se establece que una heurística de cumplir con las siguientes condiciones:

1. Debe ser capaz de proporcionar aproximaciones de alta calidad al óptimo global.
2. Debe ser robusta, es decir, ser flexible a los cambios presentados por el problema.
3. Debe poder aplicarse fácilmente a muchas instancias del problema y de hecho, poder aplicarse a nuevos problemas a pesar de que casi siempre es desarrollada para la solución de un problema específico.
4. Las heurísticas pueden ser algoritmos estocásticos pero no deben contener elementos subjetivos⁵⁴.

Existen dos tipos de heurísticas:

Heurísticas constructivas: Estas heurísticas añaden un elemento más a la solución del problema en cada paso que realizan. Son mayormente utilizadas en problemas que rutas, secuenciación, líneas de ensamble, entre otros.

Heurísticas de búsqueda local: Éstas básicamente buscan una mejor solución alrededor del punto en el que se encuentran; comenzando en una solución que no

⁵⁴ GILLI, M.; WINKER, P. Heuristic Optimization Methods in Econometrics. En: Université de Genève. Octubre 21 de 2007. 48 p.

necesariamente es aleatoria, realizan su búsqueda entre los vecinos más cercanos hasta llegar a una solución que es superior a las soluciones ofrecidas en su vecindario, esta solución es denominada óptimo local⁵⁵. Es así como existe una solución x que a medida que avanzan las iteraciones, se va moviendo alrededor del conjunto de soluciones cercanas, las cuales se denominan vecindario $N(x)$ hasta encontrar una solución x' que es mejor que la solución x inicial. Es importante resaltar que cuanto más grande sea la vecindad o vecindario, mayor calidad tendrán las soluciones ofrecidas por el programa pero también repercutirá en la utilización de más tiempo computacional⁵⁶.

Dentro de las heurísticas clásicas de búsqueda local se encuentran las denominadas Or-opt y 3-opt que son usualmente utilizadas en el VRPSD⁵⁷. Es así como a continuación se definen dichas metodologías.

- **Procedimiento k-intercambio o k-opt**

Asumiendo que se cuentan con dos soluciones denominadas T y S ; y que existe un conjunto de elementos que se encuentran en S pero que no se encuentran en T , el cual se representa mediante la forma $|S - T|$; así mismo, existen elementos que se encuentran en T pero que no se encuentran en S , siendo estos representados mediante $|T - S|$.

Si se desea realizar un intercambio simple, conocido como 1-opt, se debe iniciar el problema con la solución S , para obtener la solución T mediante la solución inicial siempre y cuando $|S - T| = |T - S| = 1$. En forma general, se

⁵⁵ GARCÍA SABATER, José Pedro; MAHEUT, Julien. Modelo y métodos de investigación de operaciones. Procedimientos para pensar. Modelado y resolución de problemas de organización industrial mediante programación matemática. En: Universitat Politècnica de València. Grupo de investigación ROGLE. Departamento de organización de empresas. Curso 2011/2012. 181 p.

⁵⁶ AHUJA, Ravindra K.; ERGUM, Özlem; ORLIN, James B.; PUNNEN, Abraham P. Estudios de técnicas de búsqueda por vecindad a muy gran escala. En: Massachusetts Institute Of Technology. Julio 1999, p 3.

⁵⁷ PELLEGRINI, Paola; BIRATTARI, Mauro. Out-of-the-Box and Custom Implementation of Metaheuristics. A Case Study: The Vehicle Routing Problem with Stochastic Demand. En: Intelligent Computational Optimization in Engineering. Springer Berlin Heidelberg. vol, 366. 2011. p 277.

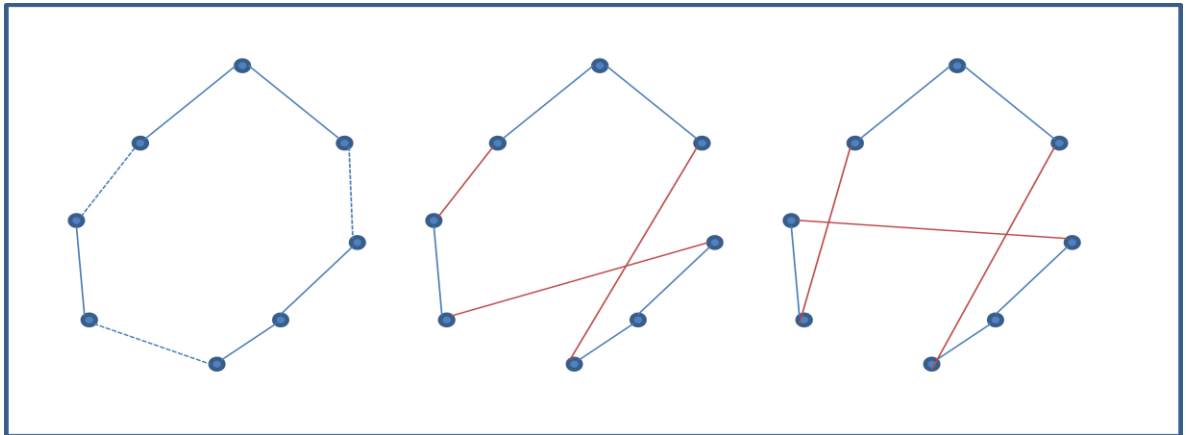
pueden realizar k - *intercambios* siempre que se cumpla que $|S - T| = |T - S| = k$ ⁵⁸.

A partir de esto, se define que el procedimiento conocido como k - *opt* consiste en eliminar k arcos de la solución inicial S para posteriormente, unir los k segmentos libres para conformar una nueva solución T .

Suponiendo que la solución consta de una trayectoria de n nodos, existen $\binom{n}{k}$ formas de eliminar k arcos y $2^{k-1}(k-1)!$ formas de construir nuevas trayectorias. Teniendo presente que un procedimiento k - *opt* tiene una complejidad computacional $O(n^k)$.

Dado que el procedimiento más utilizado para la solución del VRPSD es el 3-opt, en la siguiente figura se muestra un ejemplo de la aplicación de esta búsqueda local.

Figura 2. Intercambio 3-opt

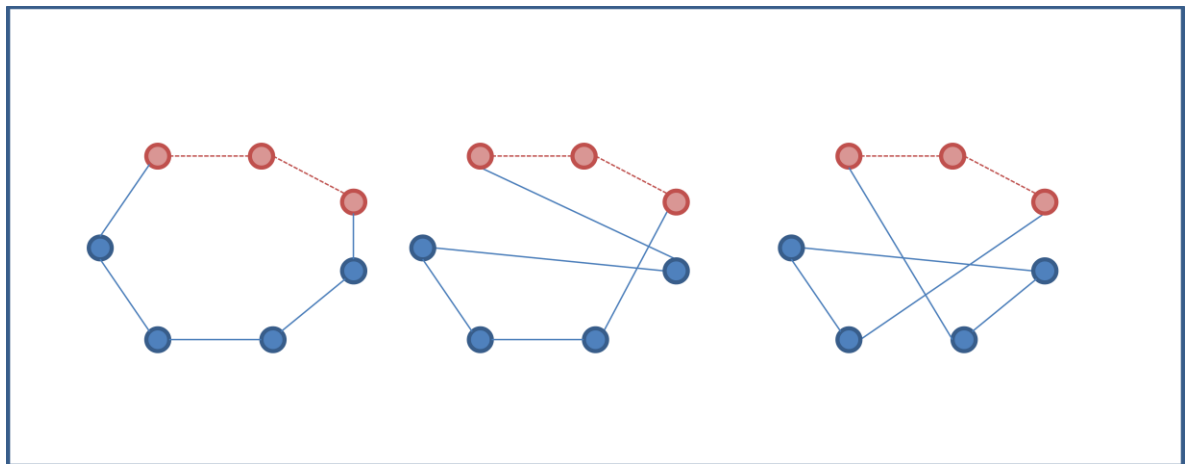


⁵⁸ AHUJA, Ravindra K.; ERGUM, Özlem; ORLIN, James B.; PUNNEN, Abraham P. Opt. Cit., p 5.

- **Procedimiento Or-opt**

Este procedimiento fue introducido por Or en 1976 y consta de la realización de cambios mediante la modificación de nodos consecutivos; esto quiere decir que el proceso comienza con la extracción de k nodos y la modificación de los mismos sin afectar el orden de estos nodos; si esto no obtiene una mejora en la solución, se realiza la extracción de $k - 1$ vértices y estos son modificados de la misma manera que como se realizó con los k nodos⁵⁹. A continuación se presenta un ejemplo de la aplicación de dicha heurística.

Figura 3. Aplicación del procedimiento Or-opt para 3 nodos.



5.3.2.2 Métodos Metaheurísticos

La mayoría de las metaheurísticas son creadas para el desarrollo de problemas de optimización combinatoria pero no están exentas a ser utilizadas en otro tipo de problemas. Este tipo de métodos de solución es enfocado en fenómenos naturales como el comportamiento de las abejas, de las hormigas o de los tornados; esto

⁵⁹ BABIN, Gilbert; DANEALU, Stéphanie; LAPORTE, Gilbert. Improvements to the Or-opt Heuristic for the Symmetric Traveling Salesman Problem. En: Cahier du GERAR no. G-2005-02. GERAD- Group for Research in Decision Analysis. Montréal, Québec, Canada. Enero 2005, p 2.

con el fin de realizar una exploración de espacio más efectiva y poder evitar la principal desventaja de las heurísticas, caer en los óptimos locales.

A continuación se presentan las metaheurísticas más comunes a la hora de solucionar problemas de optimización combinatoria.

- **BÚSQUEDA TABÚ (TS).**

Las primeras ideas de aplicación de esta metaheurística fueron desarrolladas por Glover (1986). Ésta consiste en la aplicación del proceso de búsqueda local (buscar una mejor solución en el vecindario de la solución actual) pero añadiendo a éste, una memoria a corto plazo de forma tal que cada óptimo local visitado recientemente, es guardado en una lista denominada lista tabú para garantizar que la búsqueda no recaiga en alguno de estos puntos. La introducción y salida de los óptimos locales en la lista cuando ésta ya está completamente llena, se realiza por el método PEPS (Primeros en entrar primeros en salir) de manera que cuando es añadido un óptimo local a la lista, es eliminado el valor más “viejo” de la misma.

La búsqueda Tabú también es conocida como enfoque del ascenso más empinado o descenso más suave dado que cuando realiza la búsqueda siempre busca el valor disponible que supere al punto encontrado anteriormente y en caso de no encontrarlo, se traslada al punto que menos baje en la pendiente, es decir el mayor de los menores⁶⁰.

- **TEMPLADO SIMULADO O RECOCIDO SIMULADO (SA).**

Esta metaheurística fue propuesta por Kirkpatrick, Gelatt y Vecchi (1983). El templado simulado para maximización de la función objetivo sigue la idea de buscar el pico más alto entre toda la región factible; esto se logra a través de una búsqueda local que se rige mediante una regla de selección del movimiento en la cual se acepta un nuevo punto como óptimo si éste cumple con la condición de ofrecer un resultado mayor que el ofrecido por el punto

⁶⁰ HILLIER, Op. Cit., p. 570-580.

actual, si no es así se debe determinar si es aceptado o rechazado dicho punto (esto se realiza mediante una función que determina la probabilidad de aceptación). De esta manera, la mayoría de los puntos aceptados son ascendentes, lo que aumenta la posibilidad de que gradualmente la búsqueda converja a las partes de la región factible que tienen las cumbres más altas.

La aceptación o rechazo de los puntos que no son ascendentes se da a partir de la siguiente función:

$$Prob(aceptación) = e^x \text{ donde } x = \frac{z_n - z_c}{T} \quad [1]$$

z_c = Valor de la función objetivo de la solución de prueba actual

z_n = Valor de la función objetivo de la solución del candidato actual

T = Temperatura

El parámetro T es denominado temperatura debido a la analogía con un problema físico de templado. El valor inicial de T debe ser un número relativamente grande y debe ser actualizado mediante un programa de temperaturas.

Si Número aleatorio > Prob(aceptación), se acepta un paso descendente⁶¹

- **OPTIMIZACIÓN POR ENJAMBRE DE PARTÍCULAS (PSO).**

Esta metaheurística fue introducida por James Kennedy y Russel Eberhat (1995) y se basa en el comportamiento de las abejas. Es así como un enjambre de L abejas o partículas ayuda a la determinación de la solución de un problema de optimización combinatoria. Las partículas tienen una posición que está compuesta por H dimensiones y es la que ofrece una respuesta a la función objetivo del problema; para la actualización de esta posición, cada partícula tiene su vector de velocidad que es quien determina el siguiente movimiento de la misma.

⁶¹ Ibid. p. 581-589.

El PSO se basa primordialmente en el aspecto cognitivo y social de las partículas, garantizando una búsqueda tanto local como global. Es así como cada partícula tiene una posición denominada la mejor posición de la partícula (la que hasta el momento ofrece la mejor solución a la función objetivo) y el enjambre como tal tiene la mejor posición global, la cual es la mejor solución a la función objetivo de todas las mejores soluciones ofrecidas por todas las partículas.

Al realizar la actualización de la posición actual de cada una de las partículas se requiere del vector velocidad pero éste contiene en sí los valores de la posición actual de la partícula, la mejor posición de la partícula y la mejor posición global, de forma que se incluye todo su entorno social y se garantiza una búsqueda en toda la región factible. En cada iteración se verifica si la nueva posición de la partícula es superior que la mejor posición de la misma, si es así se realiza una actualización de dicho valor y se comprueba si este nuevo valor es superior a la mejor posición global, de ser esto cierto se realiza la actualización de la nueva posición⁶².

- **ALGORITMOS GENÉTICOS**

Según la definición dada por Goldberg, los Algoritmos Genéticos son algoritmos de búsqueda basados en la mecánica de selección natural y de la genética natural. Combinan la supervivencia del más apto entre estructuras de secuencias con un intercambio de información estructurado, aunque aleatorizado, para constituir así un algoritmo de búsqueda que tenga algo de las genialidades de las búsquedas humanas.⁶³

⁶² AI, The Jin; KACHJTYICHYANUKUL, Voratas. A particle swarm optimization for the vehicle routing problema with simultaneous pickup and delivery. En: Computers & Operations Research 36. 2009. 10 p.

⁶³ GESTAL POSE, Marcos. Introducción a los algoritmos genéticos [En línea]. Departamento de Tecnologías de la información y las comunicaciones. Universidad de Coruña. [Citado el 19 de noviembre de 2014]. Disponible en: <http://sabia.tic.udc.es/mgestal/cv/aaggtutorial/tutorialalgoritmosgeneticos.pdf> OJOOOOOOO

Trasladando los conceptos biológicos al ámbito computacional, un cromosoma es un candidato a dar solución al problema, el cual es codificado como una cadena de bits. Los genes son los bits o pequeños bloques de bits que componen el cromosoma. Es así como el Algoritmo Genético, en cada nueva generación crea nuevas cromosomas utilizando bits y partes más adecuadas del progenitor. Este proceso hace que se explote eficientemente la información histórica para especular sobre nuevos puntos de búsqueda. Esperando un funcionamiento mejorado⁶⁴.

El algoritmo comienza con una población de cromosomas que se genera de manera aleatoria; todos estos son evaluados en la función fitness o función objetivo del problema. A partir de los resultados obtenidos por cada uno de los cromosomas en dicha evaluación y según la base biológica de este algoritmo, se realiza el proceso de selección de los cromosomas que deben reproducirse; esta reproducción se puede realizar mediante operaciones de cruce y de mutación. A medida que se realiza la reproducción, se crea una nueva generación de la población, la cual repetirá el proceso anterior desde la evaluación en la función fitness con el fin de encontrar un cromosoma que satisfaga todas las restricciones del problema y que cumpla el objetivo del mismo.

OPERADORES GENÉTICOS

Los tres operadores genéticos básicos de este algoritmo son la selección, reproducción o cruce y mutación.

- 1. Selección.** Este operador es utilizado para escoger qué cromosomas van a tener la oportunidad de reproducirse y qué cromosomas no las tendrán. Esta selección, al ser basada en el comportamiento biológico, busca los

⁶⁴ TOLMOS RODRÍGUEZ-PIÑERO, Piedad. Introducción a los algoritmos genéticos y sus aplicaciones. [En línea]. Universidad de Valencia. [Citado el 20 de noviembre de 2014]. Disponible en: <http://www.uv.es/asepuma/X/J24C.pdf>

cromosomas más aptos para ofrecerles una mayor oportunidad de reproducción.

Dentro de los procesos de selección se pueden encontrar la selección por ruleta y selección por torneo.

- **Selección por ruleta.** Éste es un método en el cual, a cada individuo de la población se le asigna un porcentaje (entre 0 y 1) que hace referencia a una parte de la ruleta, de forma que todos los individuos ocuparán un espacio determinado en la misma. Los cromosomas que tengan mejor resultado en la función fitness son los que tendrán una porción mayor en la ruleta, es decir un porcentaje más alto. Para seleccionar al individuo, se hace necesario girar la ruleta, lo que significa generar un número aleatorio que se encuentre entre el rango $[0,1]$ y escoger el individuo que esté situado en dicha porción de la ruleta. Aun así, está la posibilidad de que el que tenga menor probabilidad de selección sea escogido más de una vez. Y, en caso de que las probabilidades de los individuos difieran mucho, este método trae problemas puesto que se enfrascará en aquel que tenga la mayor probabilidad.
- **Selección por torneo.** Este método consiste en la comparación entre individuos. Puede ser realizado de forma determinística y de forma probabilística. Cuando la selección por torneo es determinística, lo primero que se debe definir es el valor de p , el cual indicará la cantidad de individuos a comparar; si el valor de p es 2, se toman dos individuos de la población y se selecciona el que tenga mejor valor en la función fitness. Cuando la selección es probabilística, la variante se encuentra a la hora de seleccionar al ganador del subconjunto generado por el valor p ; para este caso, se debe definir un valor p' que se encuentra en el intervalo $0,5 < p' < 1$, de forma que al generar un número aleatorio entre el rango $[0,1]$, si ese

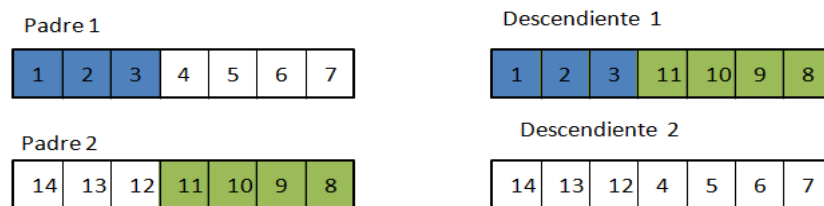
número es mayor que p' , se selecciona el cromosoma con mejor valor en la función fitness, en caso contrario se selecciona el de peor valor.

2. Cruce. Dado que ya se han seleccionado los cromosomas para la creación de la siguiente generación, se procede a realizar la reproducción de los padres. Para este proceso, se realiza la operación cruce la cual consiste en el intercambio de material genético entre los cromosomas.

Entre los métodos de cruce más empleados, se pueden encontrar el cruce de 1 punto, el cruce de 2 puntos y el cruce uniforme.

- **Cruce de 1 punto.** Ésta es una de las formas clásicas de cruce, lo que se realiza acá es la selección de un punto aleatorio dentro de los cromosomas (El mismo para ambos) y a partir de esto, el primer progenitor tendrá la información del padre 1 de inicio al punto de cruce o punto seleccionado, el resto será la información del padre 2; el segundo progenitor tendrá primero la información del padre 2 y el resto del padre 1. Para un mayor entendimiento de esta modalidad de cruce, en la figura 4, se muestra un ejemplo.

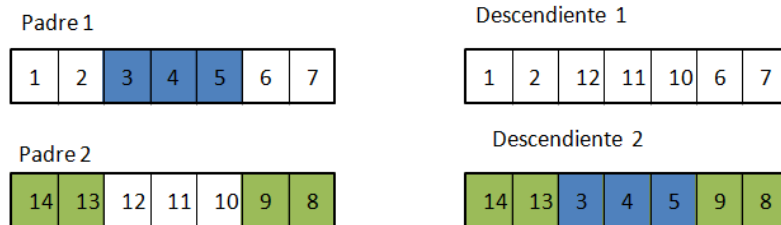
Figura 4. Cruce de 1 punto.



- **Cruce de 2 puntos.** Es la misma filosofía del cruce de 1 punto pero, para este caso se seleccionan dos puntos de cruce en los cromosomas teniendo en cuenta que dichos puntos no coincidan con los genes extremos de los

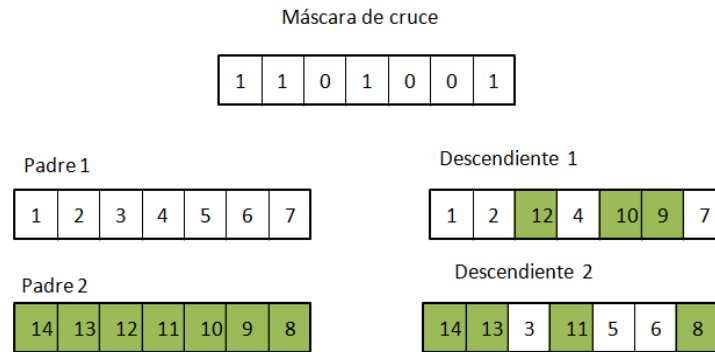
padres. Para este caso, el primer descendiente tendrá el primer conjunto de genes desde el inicio hasta el primer punto de cruce del padre 1, el segundo conjunto de genes del primer punto de cruce al segundo del padre 2 y el tercer conjunto corresponde a los genes restantes del padre 1; para el segundo descendiente, se toma primero el conjunto de inicio a punto del padre 2, el segundo conjunto corresponde al padre 1 y el último conjunto al padre 2. En la siguiente figura se muestra de forma gráfica cómo sucede este cruce.

Figura 5. Cruce de 2 puntos.



- Cruce uniforme.** Esta técnica difiere de las dos mencionadas anteriormente; aquí, todos los genes de los dos padres tienen las mismas probabilidades de ser seleccionados. Para este caso, se debe generar una máscara denominada máscara de cruce con valores binarios; para la creación del primer descendiente, se tiene que el valor de 1 representa un gen del padre 1 y el valor de 0 representa un gen del padre 2. Según la información presentada en la máscara de cruce, el primer gen del descendiente 1, será del padre 1 siempre y cuando en esa posición de la máscara se encuentre el valor de 1, en caso contrario se utilizará el gen del padre 2. Para la creación del descendiente 2 se intercambian las representaciones de forma que el valor 1 en la máscara representa un gen del padre 2 y el valor 0 representa el gen del padre 1. Para el mejor entendimiento de este cruce, en la siguiente figura se ilustra un ejemplo.

Figura 6. Cruce uniforme.



3. Mutación. La mutación genera que algunos de los genes de un cromosoma sean modificados de forma aleatoria. Los beneficios que ésta trae van encaminados a la contribución de la diversidad genética y previene que la solución tienda a algún óptimo local.

Este procedimiento usualmente va de la mano del proceso de cruce. De esta forma, los descendientes resultantes del proceso de cruce son mutados de acuerdo a una probabilidad denominada P_m . La probabilidad P_m o probabilidad de mutación toma los valores en un rango de 0 y 1, lo que indicará con qué frecuencia se realizará este proceso. Para las primeras generaciones, la mutación es una forma eficiente de realizar una búsqueda en toda la región dado que mejora la capacidad de exploración pero para la finalización de la búsqueda, la mutación puede llegar a ser destructiva por lo que se recomienda que la Probabilidad de mutación tenga valores menores que los iniciales⁶⁵.

⁶⁵ ČEPIN, Marko. Genetic Algorithm. En: Assessment of Power System Reliability. Editor: Springer London. 2011. p, 266.

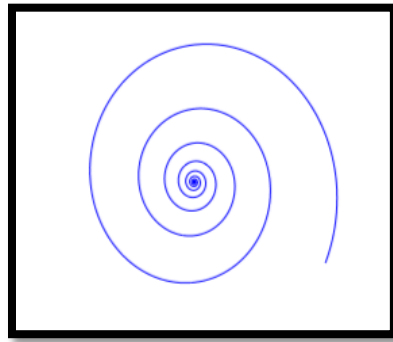
- **OPTIMIZACIÓN POR ESPIRAL.**

Es una metaheurística con un nuevo método de búsqueda multipunto desarrollada por Kenechi Tamura y Keiichiro Yasura (2011). Al igual que diversas metaheurísticas desarrolladas anteriormente, la optimización por espiral está basada en un fenómeno natural; los autores desarrollaron un método de optimización en 2-dimensiones basado en las espirales logarítmicas que comúnmente aparecen en fenómenos naturales como la cáscara de nautilus, un torbellino o un tornado.

Una de las razones por las que se desarrolló dicha metaheurística es el hecho de que las espirales logarítmicas mediante su proceso discreto ofrecen una conducta eficaz para llegar a un punto específico como se puede observar en la figura 7. Esta metaheurística ofrece una diversificación en la fase inicial y una intensificación en la fase final.

La diversificación en la fase inicial garantiza una búsqueda en toda la región de forma que se logren identificar las subregiones con altas posibilidades de obtener mejores soluciones. La intensificación en la fase final logra centrar la búsqueda en aquellas subregiones encontradas en la diversificación con el fin de determinar la mejor solución entre las mejores de dichas subregiones. Para certificar el hecho de que se realice de forma natural esta estrategia, el método de optimización por espiral establece un centro de espiral que se modifica de acuerdo al valor mínimo obtenido en cada iteración, esto con el fin de lograr una búsqueda en toda la región hasta que las variaciones del centro de la espiral sean casi nulas. La solución final será la posición del centro de la espiral al cumplirse el criterio de parada.

Figura 7. Espiral logarítmica



Fuente. Adaptado a partir de TAMURA, Kenichi y YASUDA, Keiichiro. Spiral Optimization. A new multipoint search method. En Systems, Man and Cybernetics (SMC). Octubre 2011, p 1.

A partir del desarrollo de la metaheurística en 2 dimensiones, los autores generalizaron dicho algoritmo para ofrecer una metaheurística que pueda ser aplicada a n dimensiones.

A continuación se presentan los pasos a seguir para la aplicación del algoritmo en 2 dimensiones y posteriormente se presentará la generalización del mismo.

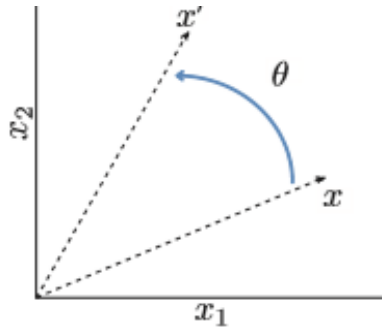
Desarrollo del algoritmo en 2 dimensiones:

Para el desarrollo del algoritmo en 2 dimensiones es necesario recordar el concepto de matriz de rotación. Una matriz de rotación es la encargada de convertir un vector a otro mediante la rotación en un eje de coordenadas. Las matrices de rotación conocidas como matrices ortogonales, tienen la característica de que el determinante de las mismas es igual a la unidad, esto demuestra el hecho de que cuando se rota un vector, la longitud del mismo luego de la rotación sigue siendo la misma. Además de esto, una matriz de rotación cumple la condición de que la transpuesta de la misma es igual a la inversa de ella.

$$R^T = R^{-1} \quad [2]$$

Como se observa en la figura 8, en un sistema de coordenadas $x_1 - x_2$, para la rotación de un vector x en un ángulo θ se tiene que:

Figura 8. Rotación de un vector x .



Fuente. Adaptado a partir de TAMURA, Kenichi y YASUDA, Keiichiro. Spiral Optimization. A new multipoint search method. En Systems, Man and Cybernetics (SMC). Octubre 2011, p 1.

$$x' = R_{1,2}^{(2)}(\theta)x \quad [3]$$

Donde:

x : es el punto inicial del vector.

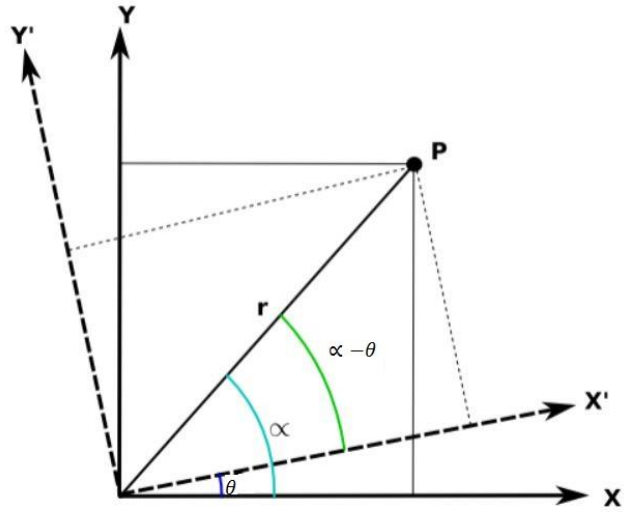
x' : es el punto obtenido por la rotación del vector.

$R_{1,2}^{(2)}(\theta)$: Es la matriz de rotación en 2 dimensiones.

$$R_{1,2}^{(2)}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \quad [4]$$

Para explicar mejor el concepto de matriz de rotación, se enuncia la forma en la que dicha matriz es obtenida.

Figura 9. Rotación alrededor del eje z.



Fuente. Adaptado de HERRA, Bernardo. Matrices de rotación.

Dado un sistema de coordenadas $x - y$, en el cual existe un punto P, si se realiza la rotación alrededor de eje z, se obtendrá un nuevo sistema de coordenadas $x' - y'$.

Se puede observar que la distancia del origen al punto P tanto para el sistema de coordenadas $x - y$ como para el sistema de coordenadas $x' - y'$ es r. Es así como las coordenadas del punto para cada sistema se pueden expresar de la siguiente manera:

$$x = r \cos(\alpha) \quad [5]$$

$$y = r \sin(\alpha) \quad [6]$$

$$x' = r \cos(\alpha - \theta) \quad [7]$$

$$y' = r \sin(\alpha - \theta) \quad [8]$$

A partir de las ecuaciones [7] y [8], se obtiene mediante aplicaciones de identidades trigonométricas lo siguiente:

$$x' = r \cos(\alpha) \cos(\theta) + r \sin(\alpha) \sin(\theta) \quad [9]$$

$$y' = r \sin(\alpha) \cos(\theta) - r \cos(\alpha) \sin(\theta) \quad [10]$$

Y reemplazando [5] y [6] en las ecuaciones [9] y [10] se logra lo siguiente:

$$x' = x \cos(\theta) + y \sin(\theta) \quad [11]$$

$$y' = -x \sin(\theta) + y \cos(\theta) \quad [12]$$

Al ser transformados a vectores y matrices se obtiene la expresión que determina la rotación para 2 dimensiones:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} * \begin{bmatrix} x \\ y \end{bmatrix} \quad [13]$$

A partir de esta definición y de [13], se puede formular un modelo que converge a un punto x_0 en un plano $x_1 - x_2$ de forma que dicho modelo desarrolla una espiral logarítmica.

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = r \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} * \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} \quad [14]$$

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = S_2(r, \theta) * x(k) \quad [15]$$

Donde k es la iteración, $0 \leq \theta \leq 2\pi$ es el ángulo de rotación alrededor del origen de cada k , $0 < r < 1$ es la tasa de convergencia de la distancia entre un punto y el origen de cada k que garantiza que exista un cambio en la longitud del vector y $S_2(r, \theta)$ es denominada matriz estable.

Para este caso, la espiral logarítmica tiene un centro fijo, denominado x_0 . Para hacer má

s flexible el algoritmo garantizando que se presente la estrategia de diversificación e intensificación, se realiza la siguiente modificación a la expresión:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = S_2(r, \theta) * x(k) - (S_2(r, \theta) - I_2)x^* \quad [16]$$

Donde I_2 es la matriz identidad en 2 dimensiones

La parte que se añade a la expresión garantiza que el centro de rotación de la espiral, el denominado x^* , sea un valor cambiante.

A continuación se presentan los pasos detallados para la realización del algoritmo de optimización en espiral en 2 dimensiones:

Primer paso: Preparación. Selección del número de espirales a utilizar teniendo en cuenta que $m \geq 2$. Establecer los parámetros $0 < \theta < 2\pi$, $0 < r < 1$ y número máximo de iteraciones $k_{m\acute{a}x}$, comenzando con $k = 0$.

Segundo paso: Inicialización. Se establecen los puntos iniciales $x_i(0)$ que pertenezcan a R^2 y que se encuentren dentro de la región factible del problema. Además de esto, se debe establecer el primer centro de espiral $x^*(0) = x_{i_g}(0)$ donde $i_g = \arg \min_i f(x_i(0))$ para $i = 1, 2, 3, \dots, m$.

Tercer paso: Actualización x_i .

$$[x_i(k+1)] = S_2(r, \theta) * x_i(k) - (S_2(r, \theta) - I_2)x^* \quad [17]$$

Para $i = 1, 2, 3, \dots, m$.

Cuarto paso: Actualización x^* .

$$x^* = x_{i_g}(k+1) \quad [18]$$

Donde $i_g = \arg \min_i f(x_i(k+1))$ para $i = 1, 2, 3, \dots, m$.

Quinto paso: Verificación del criterio de parada. Si $k = k_{m\acute{a}x}$ se termina el algoritmo, de lo contrario se establece que $k = k + 1$ y se regresa al tercer paso.

Desarrollo del algoritmo para n dimensiones:

Primer paso: Preparación. Selección del número de espirales a utilizar teniendo en cuenta que $m \geq 2$. Establecer los parámetros $0 < \theta < 2\pi$, $0 < r < 1$ y el número máximo de iteraciones $k_{m\acute{a}x}$, comenzando con $k = 0$.

Segundo paso: Inicialización. Se establecen los puntos iniciales $x_i(0)$ que pertenezcan a R^n y que se encuentren dentro de la región factible del problema. Además de esto, se debe establecer el primer centro de espiral $x^*(0) = x_{i_g}(0)$ donde $i_g = \arg \min_i f(x_i(0))$ para $i = 1, 2, 3, \dots, m$.

Tercer paso: Actualización x_i .

$$[x_i(k + 1)] = S_n(r, \theta) * x_i(k) - (S_n(r, \theta) - I_n)x^* \quad [19]$$

Para $i = 1, 2, 3, \dots, m$.

Cuarto paso: Actualización x^* .

$$x^* = x_{i_g}(k + 1) \quad [20]$$

Donde $i_g = \arg \min_i f(x_i(k + 1))$ para $i = 1, 2, 3, \dots, m$.

Quinto paso: Verificación del criterio de parada. Si $k = k_{m\acute{a}x}$ se termina el algoritmo, de lo contrario se establece que $k = k + 1$ y se regresa al tercer paso.

Se puede observar que los pasos a seguir para el desarrollo del algoritmo en 2 dimensiones y el desarrollo del algoritmo en n dimensiones son iguales excepto por:

1. Los vectores x_i pertenecen a R^n .
2. La matriz Identidad I pertenece a R^n .
3. La matriz de rotación será $R^{(n)}(\theta_{1,2}, \theta_{1,3}, \dots, \theta_{n-1,n})$

Para la determinación de dicha matriz de rotación se debe tener en cuenta:

La rotación sólo se puede realizar en un plano, es decir, sólo se puede realizar una rotación con respecto a dos dimensiones o dos ejes $x_i - x_j$ que pertenecen a las n dimensiones del problema. Las demás dimensiones presenten se mantienen constantes. El conjunto de todas las matrices de rotación de tamaño n forman un grupo conocido como **grupo especial ortogonal**.

La matriz de rotación de n dimensiones es la permutación de cada una de las posibles rotaciones presentes en ese problema o de la permutación del grupo especial ortogonal. Se define mediante la siguiente ecuación:

$$\begin{aligned}
 R^{(n)}(\theta_{1,2}, \theta_{1,3}, \dots, \theta_{n,n-1}) & \\
 & := R_{n-1,n}^{(n)}(\theta_{n-1,n}) * R_{n-2,n}^{(n)}(\theta_{n-2,n}) R_{n-2,n-1}^{(n)}(\theta_{n-2,n-1}) * \dots * R_{2,n}^{(n)}(\theta_{2,n}) \\
 & * \dots * R_{2,3}^{(n)}(\theta_{2,3}) R_{1,n}^{(n)}(\theta_{1,n}) * \dots * R_{1,3}^{(n)}(\theta_{1,3}) R_{1,2}^{(n)}(\theta_{1,2}) \quad [21] \\
 & = \prod_{i=1}^{n-1} \left[\prod_{j=1}^i R_{n-i,n+1-j}^{(n)}(\theta_{n-i,n+1-j}) \right]
 \end{aligned}$$

5.4 VEHICLE ROUTING PROBLEM (VRP)

El problema de ruteo de vehículos (VRP) surge a partir del problema del agente viajero (TSP) en el cual un vendedor parte de un punto de origen y debe visitar a cada cliente una sola vez, retornando finalmente al punto de partida. Se considera que los primeros estudios sobre el TSP fueron realizados por Menger (1932)⁶⁶, sin embargo, formalmente se introdujo este concepto por Flood (1956)⁶⁷.

⁶⁶ PARLAKTUNA, Osman; IPAHI OGLU, Aydin S.; YAZICI, Ahmet. A VRP-Based Route Planning for a Mobile Robot Group. En: Turk J Elec Engin, VOL.15, NO.2, 2007, 12p.

⁶⁷ ROCHA, Op. Cit., p. 35 - 55.

A partir de una generalización del TSP realizada por Dantzig y Ramser (1959) surge el primer problema de VRP, al modelar matemáticamente el abastecimiento de gasolina a estaciones de servicio por medio de una flota de vehículos.⁶⁸

Es entonces como surgen diversas clasificaciones del VRP, las cuales de acuerdo a las características de los componentes del problema se pueden clasificar en homogéneo o heterogéneo.

Antes de establecer la clasificación, se mencionará el **Capacitated Vehicle Routing Problem (CVRP)**, el cual es considerado el problema de ruteo base para mucho de los tipos de VRP presentes en las dos clasificaciones. En éste se restringe la capacidad de los vehículos teniendo en cuenta que éstos sólo deben visitar a un cliente una vez y deben lograr satisfacer la demanda total al menor costo posible⁶⁹. Es entonces como se pasa de un TSP generalizado a un problema de ruteo de vehículos cada vez más específico.

1. **VRP HOMOGÉNEO:** Se denomina de esta forma debido a que presenta una característica común en alguno de los componentes del problema como distancias, ventanas de tiempo, capacidad⁷⁰.

Dentro de esta clasificación se pueden encontrar tres tipos de VRP que a su vez se pueden subdividir en más tipos debido a pequeñas variaciones en los componentes o debido a híbridos que surgen de la combinación de diversos VRP presentes en este grupo. A continuación se presenta una definición general de cada uno.

- **Vehicle Routing Problem with Time Windows (VRPTW):** El primer estudio realizado sobre este tema fue hecho por Pullen y Webb (1967)⁷¹; este

⁶⁸ RODRIGUEZ PÉREZ, Jorge. Caracterización, modelado y determinación de las rutas de la flota en una empresa de Rendering. En: E-Reding. Trabajos y proyectos fin de estudios de la E.T.S.I. Máster en Organización Industrial y Gestión de Empresas. 2012, 700 p.

⁶⁹ QI, Chengming; CUI, Shoumei; SUN, Yunchuan. A two-Stage Hybrid Ant Colony Algorithm for the CVRP. En. International Conference on Computational Intelligence and Security. 2009 5 p.

⁷⁰ ROCHA, Op. Cit., p. 35 - 55.

problema tiene como característica fundamental la limitación del tiempo de servicio así como el tiempo de llegada y salida del vehículo a cada uno de los nodos o clientes; es así como se añade la denominada ventana de tiempo a cada uno de los clientes presentes en el problema. Si el vehículo llega al cliente antes de lo previsto, debe esperar que la ventana de tiempo de ese cliente sea abierta, es decir, que llegue el momento especificado para atender dicho cliente; a el vehículo no se le permite llegar después de que se ha cerrado la ventana de tiempo⁷². Un aspecto importante a resaltar del VRPTW es que busca, principalmente, la minimización del número rutas, vehículos y tiempos de viaje; y como segunda instancia la minimización del costo total del viaje⁷³.

- **Vehicle Routing Problem with split deliveries (SDVRP):** Esta extensión del VRP elimina una de las restricciones básicas del problema, la cual enuncia que la demanda de cada uno de los clientes no debe ser superior a la capacidad de los vehículos; es así como puede ocurrir que un cliente deba ser visitado por varios vehículos con el fin de suplir su demanda al menor costo posible⁷⁴. Dror y Trudeau (1989) comenzaron a estudiar y dar solución al SDVRP mediante una búsqueda local⁷⁵.
- **Vehicle Routing Problem with Backhauls (VRPB):** Como su nombre lo indica, este tipo de VRP permite que los vehículos realicen viajes de regreso, es decir que recogen mercancía de los nodos o clientes y se dirigen hacia el depósito con ella. En este problema se establece que las devoluciones, viajes de regreso o Backhauls se comienzan a realizar cuando ya se han finalizado

⁷¹ CORDEAU, J.-F.; DESAULNIERS, Guy; DESROSIERS, Jacques; et al. The VRP with Time Wondows. En: GERAD. Febrero 1999. Revisado Junio 2000. 38 p.

⁷² QI, Chengming; SUN, Yunchuan. An Improved Ant Colony Algorithm for VRPTW. En: International Conference on Computer Science and Software Engineering. 2008, p. 455-458.

⁷³ QI, Chengming; CUI, Shoumei; SUN, Yunchuan. Using ant colony system and local search methods to solve VRPTW. EN: Pacific-Asia Workshop on Computational Intelligence and Industrial Ap.lication. 2008, p. 478-482.

⁷⁴ ARCHETTI, C.; SPERANZA, M.G.; HERTZ, A. A tabu search algorithm for the Split delivery vehicle routing problema. En: Transportation science. Vol 40. No. 1. Frebero 2006, oo 64-73. (ISSN 0041-1655).

⁷⁵ SPERANZA, M. Gracia. The Split delivery vehicle routing problema. En: University of Brescia- Italy. 1 p.

todas las entregas a los clientes⁷⁶. Este VRP está muy relacionado con el ***Vehicle Routing Problems with Pickups and Deliveries (PDPVRP)*** el cual especifica que los vehículos pueden hacer tantas entregas como devoluciones de mercancía; en este caso, la mercancía devuelta por el cliente queda en el mismo vehículo por lo que no se debe dirigir a un depósito⁷⁷; Golden, Baker, Alfaro y Schaffer (1985) realizaron los primeros estudios sobre VRPB; para ellos los clientes a los que se les entregaba la mercancía eran denominados linehaul customers y los que generaban los regresos al depósito eran denominados backhaul customers⁷⁸.

2. VRP HETEROGÉNEO: Corresponde a un conjunto de variantes del VRP que se caracterizan por la desigualdad de sus componentes. A continuación, se presenta la clasificación de los diversos tipos de problemas pertenecientes a este grupo.

- **Vehicle Routing Problem Heterogeneous Fleet (VRPHF):** Esta variante del VRP se caracteriza porque los vehículos utilizados difieren en diversos atributos como la capacidad, velocidad, costos, entre otros.⁷⁹
- **Periodic Vehicle Routing Problem (PVRP):** Este problema se diferencia del VRP clásico debido a que en este el periodo de planeación deja de ser de un día para pasar a ser de varios. Así mismo, no necesariamente el vehículo

⁷⁶ RODRIGUEZ PÉREZ, Jorge. Caracterización, modelado y determinación de las rutas de la flota en una empresa de Rendering. En: E-Reding. Trabajos y proyectos fin de estudios de la E.T.S.I. Máster en Organización Industrial y Gestión de Empresas. 2012, 700 p.

⁷⁷ BENAVENTE, Magdalena; BUSTOS, Jaime. Estado del arte en el problema de ruteo de vehículos (VRP). En: Laboratorio de Ingeniería aplicada, departamento de ingeniería de sistemas, Universidad de La Frontera. 13 p.

⁷⁸ NAGY, Bábor; SALHI, Saïd. Heuristic Algorithms for single and multiple depot vehicle routing problems with Pickups and deliveries. En: Working paper series. No.42. Canterbury Business School. Noviembre 2003. 28 p.

⁷⁹ CERANOGLU, Ahmet; DUMAN, Ekrem. VRP12 (vehicle routing problem with distances one and two) with side constraints. En: International Journal of Production Economics. Agosto 2013. Vol. 144. p. 461-467.

debe retornar al depósito el mismo día que salió, por lo que se puede especificar dicho periodo de tiempo.⁸⁰

- **Vehicle Routing Problem Multiple Depots (Multi-depot VRP):** Representa una generalización del CVRP, caracterizada por la existencia de múltiples depósitos a fin de servir a los clientes. Además, cada depósito cuenta con una serie de vehículos los cuales luego de su recorrido retornan a sus respectivos lugares.⁸¹
- **Vehicle Routing Problem Multiple Trips (Multi-trip VRP):** Esta extensión del VRP se caracteriza porque cada vehículo puede realizar varias rutas dentro del mismo periodo de planeación. Para esto, luego del establecimiento de las rutas, se realiza la asignación de las mismas a la disponibilidad de la flota de vehículos.⁸²
- **Multi Capacity Vehicle Routing Problem (MCVRP):** En este problema se realiza el transporte y entrega de múltiples productos gracias a que los vehículos presentan varios compartimientos dentro de los cuales se pueden almacenar las diversas mercancías.⁸³
- **Multi Objective Vehicle Routing Problem (MOVRP):** En este tipo de problemas se desarrolla la optimización simultánea de varios objetivos relevantes del VRP como la distancia recorrida, el número de vehículos requeridos, el costo de transporte, la satisfacción del cliente, entre otros. Es importante resaltar que la determinación de estos objetivos varía de acuerdo a las intenciones del investigador.⁸⁴
- **Stochastic Vehicle Routing Problem (SVRP):** El problema del ruteo de vehículos estocástico se caracteriza porque alguno de sus componentes presenta un comportamiento aleatorio. De acuerdo al parámetro que es

⁸⁰ MOLINA, José; EGUÍA, Ignacio. PROBLEMA DE OPTIMIZACION DE RUTAS DE VEHICULOS CON ASPECTOS MEDIOAMBIENTALES. Tesis de maestría en Organización Industrial y Gestión de Empresas. Sevilla. Universidad de Sevilla. Noviembre 2013.

⁸¹ Ibid

⁸² ROCHA, Op. Cit., p. 41.

⁸³ RODRIGUEZ, Op. Cit., p. 17

⁸⁴ ROCHA, Op. Cit., p. 41.

aleatorio surgen tres variaciones para este problema: **VRPSD**, la demanda es una variable aleatoria; **VRPSTT**, los tiempos de servicio y de viaje son variables aleatorias; **SVRP-SN**, la presencia de los clientes es una variable aleatoria.⁸⁵

5.4.1 Vehicle Routing Problem with Stochastic Demands (VRPSD). El VRPSD es un problema de complejidad computacional *NP-HARD*, en el cual existen n clientes con demandas estocásticas, lo que quiere decir que no se conoce la cantidad exacta que demandará cada uno; sólo se conoce la distribución estadística que sigue dicha demanda. Solamente cuando el vehículo llega al nodo a servir al cliente, se conoce la demanda real del mismo. Este problema cuenta con las siguientes restricciones⁸⁶:

- Todas las rutas comienzan y terminan en el depósito.
- Cada vehículo tiene una capacidad máxima Q y es la misma para todos los vehículos.
- Cada cliente tiene una demanda estocástica que debe ser satisfecha.
- El vehículo no puede ir dos veces al mismo nodo, si lo hace debe incurrir en una penalización.

Sea $G = \{V, A, C\}$ un grafo completo donde $V = \{0, 1, 2, \dots, n\}$ es un conjunto de nodos o clientes donde el valor 0 equivale al depósito, $A = \{(i, j) : i, j \in V, i \neq j\}$ es el conjunto de arcos que unen los nodos, y $C = \{c_{ij} \in V, i \neq j\}$ es una matriz que contiene los costos de viajar del nodo i al nodo j . Esta matriz es simétrica, lo que indica que cada arco tiene un costo tal que $c_{ij} = c_{ji}$ y satisface la desigualdad triangular $c_{ij} \leq c_{ik} + c_{kj}$.

⁸⁵ MOLINA, Op. Cit., p. 21.

⁸⁶ JUAN, Ángel A, et al. Applying Simulation and Reliability to Vehicle Routing Problems with Stochastic Demands. En: XI Conference of the Italian Association for Artificial Intelligence (AIIA09), Reggio-Emilia (Italia). 2009. P. 2

El vehículo con la capacidad Q definida, debe satisfacer la demanda de los clientes teniendo en cuenta que se debe minimizar el costo total esperado del viaje. Dada la demanda estocástica de los clientes $\xi_i, i = 1, \dots, n$ se tiene que dicha demanda sigue la siguiente distribución de probabilidad discreta y uniforme $p_{ik} = Prob(\xi_i = k), k = 0, 1, 2, \dots, K \leq Q$ ⁸⁷.

5.4.1.1 Políticas de servicio

Según Secomandi⁸⁸ (2001), en la literatura se han trabajado dos enfoques de servicio para la solución del VRPSD. El primer enfoque es un enfoque estático denominado enfoque de optimización a priori y el segundo enfoque es el dinámico o enfoque de reoptimización (Dror et al. 1989 y Dror 1993).

A. ENFOQUE A PRIORI

Bajo este enfoque se establecen rutas antes de conocer las demandas de los clientes, de forma que el vehículo distribuye la mercancía siguiendo la ruta de clientes establecida antes de iniciar el recorrido. Dado que la demanda de los clientes sólo se sabe al momento de atenderlos, se puede presentar un fallo en la ruta, lo cual significa que el vehículo no tiene la mercancía necesaria para satisfacer la demanda del nodo actual; para esto, se presentan distintas estrategias que evitan la modificación de la secuencia planteada de forma a priori.

Considerando una ruta a priori $S = (1, 2, 3, \dots, n)$, donde los clientes han sido ordenados por nodos, el vehículo visitará a los clientes según se ha definido en la ruta a priori; en el instante en que el vehículo presente una falla en la ruta, volverá al depósito y posteriormente atenderá al cliente no satisfecho y continuará las visitas según lo planteado en la ruta, el vehículo también puede volver al depósito para hacer rellenos tempranos o un reabastecimiento preventivo y después,

⁸⁷ BIANCHI, Leonora; et al. Metaheuristics for the Vehicle Routing Problem with Stochastic Demands. En: Parallel Problem Solving from Nature-P.SN. Enero, 2004. vol. 8, 450 p.

⁸⁸ SECOMANDI, Nicola. A rollout policy for the vehicle routing problem with stochastic demands. En: Operations Research. 2001. Vol. 49, no. 5, p. 796–802.

continuar el recorrido según lo establecido a priori. Esta política de servicio fue utilizada por primera vez por Bertsimas⁸⁹ (1992) y posteriormente mejorada por Bertsimas, et al. (1995).

Así mismo, se han presentado distintos estudios sobre este tipo de política. Entre estas se puede mencionar las realizadas por Ak y Erera (2007)⁹⁰ quienes aplican la estrategia denominada vehículos emparejados para el ruteo de vehículos con demandas estocásticas. Además, se destaca la definición de reabastecimiento preventivo presentada por Bianchi⁹¹ (2004) el cual considera que “Algunas veces la mejor opción es elegir el reabastecimiento, así el vehículo no este vacío, o si su capacidad es mayor que la demanda esperada del próximo cliente programado”⁹²; esta política de reabastecimiento preventivo es también denominada política mixta debido a que combina el enfoque a priori con el enfoque dinámico.

B. ENFOQUE DE REOPTIMIZACIÓN

En este enfoque, el enrutamiento es denominado dinámico dado que se establece a medida que se ofrece el servicio, es decir, que en el momento en que se atiende a un cliente se toma la decisión de hacia dónde se dirige el vehículo. Es así como el vehículo puede optar por ir a atender a otro cliente o dirigirse al depósito a realizar un reabastecimiento.

Para este caso, se pueden presentar distintas políticas de servicio, entre las que se encuentran la política de entrega completa y la política de entrega dividida⁹³.

La política de entrega completa supone que $\xi_i \leq Q$; esto debido a que cada vez que el vehículo llega al nodo de un cliente, éste debe satisfacer toda la demanda que se exige en ese nodo, lo que hace necesario que se añada una nueva

⁸⁹ Ibid.

⁹⁰ AK, Op. Cit., p. 222-237.

⁹¹ BIANCHI, Op. Cit., p. 450-480.

⁹² BIANCHI, Op. Cit., p. 450.

⁹³ DROR, Mosher; LAPORTE, Gilbert y TRUDEAU, Pierre. Vehicle Routing with Stochastic Demands: Properties and Solution Frameworks. En: Transportation Science. Agosto, 1989, vol. 23, no.3, p. 168.

suposición que garantice que esto siempre se va de cumplir. En cuanto a la política de entregas divididas, se elimina la restricción de que el cliente sólo puede ser visitado una vez por un solo vehículo, lo que da lugar a que la demanda que exige el cliente pueda ser satisfecha por varios vehículos o por varias visitas de un mismo vehículo⁹⁴.

5.4.1.2 Elección de la política de servicio

Según la literatura, el enfoque de reoptimización o enfoque dinámico es el más utilizado para el desarrollo de problemas de ruteo de vehículos (VRP)⁹⁵ dado que éste ofrece soluciones de mayor calidad; Sin embargo, cuando se establece un problema de ruteo de vehículos con demandas estocásticas (VRPSD), este enfoque tiende a presentar dificultades que se ven reflejadas en esfuerzos y costos computacionales demasiado grandes, esto debido a que, dada la complejidad del problema, muchas de las instancias del mismo deben resolverse exponencialmente. Es así como los algoritmos de solución basados en el enfoque de reoptimización para el VRPSD son utilizados para casos en los que el tamaño del problema es pequeño⁹⁶.

En cuanto al enfoque a priori, se logra tener esfuerzos y costos computacionales menores; además de esto, utilizando la estrategia de reabastecimiento preventivo, se evita que el vehículo presente fallos en la ruta que generan mayores costos⁹⁷.

Es así como, en esta investigación se estudia el VRPSD con el enfoque a priori y la estrategia de reabastecimiento preventivo.

⁹⁴ Ibid.

⁹⁵ Ibid.

⁹⁶ FAN, Jianhua; et al. Op. Cit., p. 1689.

⁹⁷ BIANCHI, Op. Cit., p. 450.

5.4.1.3 Modelamiento matemático del VRPSD

Dado que el objetivo principal del VRPSD es la minimización de los costos esperados del viaje total; se hace necesaria la definición de cada uno de los mismos presentes en dicho problema.

Con base en lo anterior y empleando el enfoque a priori con reabastecimiento preventivo, se tienen los siguientes costos:

- Un costo de ir del nodo o cliente i al nodo j según la ruta a priori definida (c_{ij}).
- Los costos pertenecientes a la política de servicio. Para este caso, existirá un costo por el reabastecimiento preventivo y un costo por los fallos en la ruta.

Dada las condiciones del problema, el vehículo partirá su recorrido según la ruta a priori definida; después de servir completamente al cliente j , el vehículo tendrá una carga q menor a su capacidad total Q , que presentará un costo $f_j(q)$ definido como el costo esperado total desde el nodo j hacia adelante. Si L_j se define como el conjunto de todas las posibles cargas q que puede tener el vehículo después de atender al cliente j tal que $q \in L_j$, entonces $f_j(q)$ satisface que:

$$f_j(q) = \min[f_j^p(q), f_j^r(q)]^{98} \quad [23]$$

Donde:

$$f_j^p(q) = c_{j,j+1} + \sum_{k:k \leq q} f_{j+1}(q-k) p_{j+1,k} + \sum_{k:k > q} [2c_{j+1,0} + f_{j+1}(q+Q-k)] p_{j+1,k} \quad [24]$$

Y

$$f_j^r(q) = c_{j,0} + c_{0,j+1} + \sum_{k=1}^K f_{j+1}(Q-k) p_{j+1,k} \quad [25]$$

⁹⁸ BIANCHI, Leonora, et al. Hybrid Metaheuristics for the Vehicle Routing Problem with Stochastic Demands. En: Journal of Mathematical Modelling and Algorithms. 2006, 94 p.

Con las siguientes condiciones de frontera:

$$f_n(q) = c_{n,0} \quad q \in L_n \quad [26]$$

Tal que, $f_j^p(q)$ representa el costo esperado de proceder directamente al siguiente nodo, $f_j^r(q)$ representa el costo del abastecimiento preventivo. El término $f_j^p(q)$ definido en la ecuación [24] está conformado por el costo de ir del nodo j al nodo $j + 1$; el costo $f_{j+1}(q - k)$ que representa el costo esperado total desde el nodo $j + 1$ hacia adelante multiplicado por la probabilidad $p_{j+1,k}$ que representa la probabilidad de que en el nodo $j + 1$ el cliente pida k cantidad de mercancía; y el término $\sum_{k:k>q} [2c_{j+1,0} + f_{j+1}(q + Q - k)]p_{j+1,k}$ que representa el costo por falla en la ruta, el cual está conformado por $2c_{j+1,0}$ el cuál es el costo de ir del cliente $j + 1$ al depósito y del depósito al cliente $j + 1$, $f_{j+1}(q + Q - k)$ es el costo total esperado desde el nodo $j + 1$ hacia adelante y $p_{j+1,k}$ que representa la probabilidad de que en el nodo $j + 1$ el cliente pida k cantidad de mercancía. El término $f_j^r(q)$ definido en la ecuación [25] está conformado por el costo de ir del nodo j al depósito, el costo de ir del depósito al nodo $j + 1$ y el costo $f_{j+1}(q - k)$ que representa el costo de esperado total desde el nodo $j + 1$ hacia adelante multiplicado por la probabilidad $p_{j+1,k}$ que representa la probabilidad de que en el nodo $j + 1$ el cliente pida k cantidad de mercancía. Por último, la ecuación [26] muestra que cuando el vehículo ha visitado al último cliente, sin importar la carga que tenga, el costo será el costo de ir del cliente n al depósito.

Para el desarrollo del abastecimiento preventivo, existirá un valor h_j que representa el umbral para la carga q que tiene el vehículo después de servir al cliente j . De forma que si $q < h_j$, el vehículo volverá al depósito a realizar el abastecimiento.

6. EXPERIMENTACIÓN

6.1 FUNCIONES DE PRUEBA

Para comprobar el correcto funcionamiento del algoritmo de la metaheurística Optimización por Espiral, desarrollado por medio del software computacional MATLAB; se procede a realizar unas pruebas cuyas funciones objetivos están determinadas por las funciones planteadas en el artículo desarrollado por Kenichi Tamura y Keiichiro Yasuda⁹⁹. Los resultados de dicho proceso, se encuentran en el Anexo A.

6.2 APLICACIÓN DE LA METAHEURÍSTICA OPTIMIZACIÓN POR ESPIRAL A UN PROBLEMA DE TSP

Luego de la revisión del software MATLAB y del estudio de la metaheurística optimización por espiral, se procedió a solucionar un problema propuesto en la literatura con el objetivo de afianzar los conocimientos acerca del funcionamiento del algoritmo y facilitar así su posterior aplicación al problema del VRPSD. De esta forma se seleccionó el ejercicio del problema del agente viajero (Traveling Salesman Problem, TSP), propuesto en el libro “Introducción a la investigación de operaciones”¹⁰⁰.

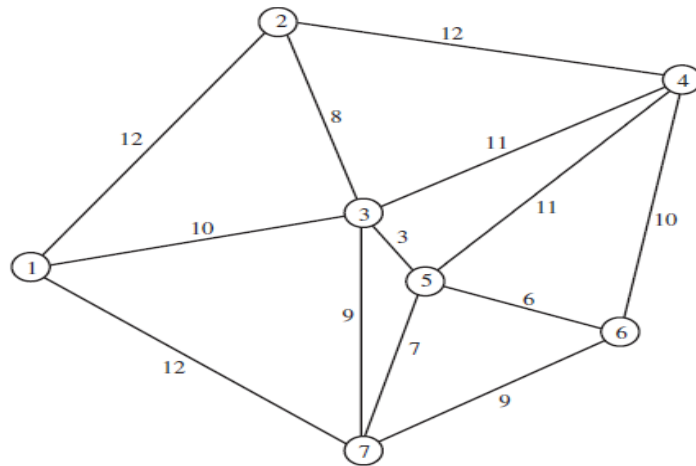
En la figura 10 se presenta el problema correspondiente. Como se puede observar, el diagrama está compuesto por siete ciudades de las cuales la número uno corresponde a la residencia del agente viajero. La esencia del problema consiste en encontrar la ruta que minimiza la distancia recorrida por parte del agente, de forma de que visite tan sólo una vez a cada una de las otras ciudades antes de volver a su lugar de residencia. Así mismo, los valores de las aristas que

⁹⁹ TAMURA, Op. Cit, p. 4-6.

¹⁰⁰ HILLIER, Op. Cit., p. 567-570.

unen a cada par de ciudades corresponden a la distancia entre cada una de ellas.¹⁰¹

Figura 10. Ilustración del problema del agente viajero.



Fuente. Adaptado de HILLIER, Frederick S.; LIEBERMAN, Gerald J. Introducción a la investigación de operaciones. 9° Ed. Ciudad de México, México: Mc Graw Hill. 2010, 978 p.

Teniendo en cuenta la optimización por espiral, cada solución se codifica como la permutación entera $Y_i = (y_{i1}, y_{i2}, y_{i3}, \dots, y_{in})$, donde i representa a la espiral i -ésima y n representa el número de clientes.¹⁰²

Transformación de la permutación de clientes a valores de posición

Esencialmente lo que se realiza es la conversión de la permutación de clientes (valores discretos) en valores de posición continuos dentro del espacio de solución. La conversión se realiza usando la siguiente ecuación:

$$x_{ij} = x_{min} + \frac{(x_{max} - x_{min})}{n[y_{ij} - 1 + rand()]} \quad [27]$$

¹⁰¹ Ibid.

¹⁰² GEETHA, S; GANESAN, Poonthalir ;VANATHI, P T. A Hybrid Particle Swarm Optimization with Genetic Operators for Vehicle Routing Problem. En: Journal of Advances in Information Technology. 2010. Vol. 1, no. 4, p. 181–188. DOI 10.4304/jait.1.4.181-188.

Donde:

y_{ij} : j-ésima dimensión de la solución i-ésima.

x_{ij} : j-ésima dimensión de la partícula i-ésima en el espacio de solución.

x_{min} : Valor mínimo de frontera del valor posición de la espiral en el espacio de solución.

x_{max} : Valor máximo de frontera del valor posición de la espiral en el espacio de solución.

$rand()$: número aleatorio distribuido uniformemente en [0,1].

n : número de clientes.

A fin de ilustrar la transformación de la permutación de clientes a valores de posición se realiza el siguiente ejemplo tomando los valores de x_{min} , x_{max} y n asumidos por Geetha et al.¹⁰³

Tabla 1. Parámetros iniciales para el desarrollo del problema de TSP.

Parámetros	
x_{min}	0.1
x_{max}	0.7
N	5
Permutación de clientes	$Y_i = (1,2,3,4,5)$

De esta manera el vector X_i obtenido de la aplicación de la ecuación 27, se presenta en la tabla 2.

¹⁰³ Ibid

Tabla 2. Decodificación de la solución codificada.

Dimensión j	1	2	3	4	5
Solución codificada (Y_i)	1	2	3	4	5
Vector posición (X_i)	0.2913	0.3632	0.4098	0.5278	0.6547

Transformación de los valores de posición a la permutación de clientes

Se convierten de nuevo los valores continuos de posición del vector X_i en valores discretos del vector solución Y_i . Para esto se utiliza la regla ROV (Rank Order Value), creada por Bo, Wang y Jing¹⁰⁴ en la cual se asigna la valoración de 1 al menor valor dentro de cada espiral (Smallest Position Value, SPV). Posteriormente se asigna la valoración de 2 al segundo menor valor posición y así sucesivamente para los demás valores de la espiral.

A modo de ejemplo, asumiendo de nuevo que el número de clientes es de cinco y que previamente se ha realizado la rotación de cada uno de los puntos por medio de la metaheurística optimización por espiral. Se presenta el vector Y_i al aplicar la regla ROV, en la tabla 3.

Tabla 3. Codificación del vector posición obtenido.

Dimensión j	1	2	3	4	5
Vector posición (X_i)	0.6654	0.5265	0.3134	0.1199	0.2653
Solución codificada (Y_i)	5	4	3	1	2

¹⁰⁴ LIU, Bo, WANG, Ling y JIN, Yi-hui. An effective hybrid PSO-based algorithm for flow shop scheduling with limited buffers. En: Computers & Operations Research. 2008. Vol. 35, p. 2791–2806. DOI 10.1016/j.cor.2006.12.013.

Desarrollo del algoritmo

De esta manera, el algoritmo desarrollado para resolver el problema propuesto se presenta a continuación:

1. Crear rutas a priori de forma aleatoria.(Cada ruta estará almacenada en una espiral)
2. Repetir mientras se cumpla el criterio de parada
3. Establecer el centro de las espirales
 - 3.1 Evaluar cada espiral en la función objetivo
 - 3.2 Si el valor de la función objetivo es menor que la solución anterior
 - 3.2.1 Establecer la actual espiral como el centro de las espirales
4. Codificar cada vector al espacio continuo
5. Rotar cada espiral codificada respecto al centro
6. Transformar de nuevo cada vector al espacio discreto
7. Regresar al paso 2
8. Fin
9. Establecer el centro de las espirales como la mejor solución global

Así mismo, los parámetros empleados para el desarrollo del problema en cuestión se presentan en la tabla 4:

Tabla 4. Parámetros utilizados para la resolución del problema del TSP.

Parámetros ¹⁰⁵	
x_{min}	0.1
x_{max}	0.7
Clientes	7
Espirales	20
Ángulo	$\pi/10$
Radio	0.99

Finalmente, se llevó a cabo la comparación de los resultados alcanzados por medio del algoritmo con los presentados en el libro y se encontró que la ruta óptima obtenida: 1-2-4-6-7-5-3-1, minimiza la distancia recorrida a 63 y corresponde a la ruta óptima del problema propuesto.

¹⁰⁵ El análisis del valor óptimo de los parámetros se realizará para el problema del VRPSD.

7. CÁLCULO DE LA FUNCIÓN OBJETIVO

Dado que el VRPSD tiene una naturaleza estocástica, los costos de transporte varían según la aleatoriedad de las demandas de los clientes, por lo tanto, es necesario evaluar el costo promedio para una ruta a priori para varios escenarios, en los que las demandas de los clientes varíen según su distribución de probabilidad.¹⁰⁶.

Para una mayor comprensión en el cálculo de la función objetivo, se expondrá mediante un ejemplo sencillo la manera en la que se realiza dicho procedimiento, teniendo en cuenta las siguientes consideraciones:

- Sólo existe un vehículo.
- Capacidad del vehículo $Q = 15$.
- Las coordenadas de los clientes están definidas por $Depósito = [0\ 0]$, $A = [1\ 1]$, $B = [4\ 4]$, $C = [3\ 3]$ y $D = [2\ 2]$.
- El costo de ir del depósito o del cliente i al cliente j o al depósito, está dado por la distancia euclidiana entre los dos puntos.
- Ruta a priori $R = Depósito - A - D - C - B - Depósito$.
- Conjunto de demandas de cada cliente. $A = [10, 8, 3]$, $B = [4, 7, 10]$, $C = [2, 10, 6]$ y $D = [1, 5, 6]$.
- Un número total de escenarios de $E = 4$.
- El valor del umbral para cada uno de los clientes es el promedio de sus demandas. $h_A = 7$, $h_B = 7$, $h_C = 6$ y $h_D = 4$.

Se procede a definir el costo presentado en cada uno de los escenarios.

¹⁰⁶ TAN, K.C.; CHEONG, C.Y.; GOH, C.K. Solving multiobjective vehicle routing problem with stochastic demand via evolutionary computation. En: European Journal of Operational Research. 2007, vol, 177, no.2, p 821.

- Para cada uno de los escenarios, se definen de forma aleatoria las demandas de cada uno de los clientes de acuerdo a sus intervalos de demandas.

$$E1 = \{A = 3, D = 1, C = 2, B = 4\}$$

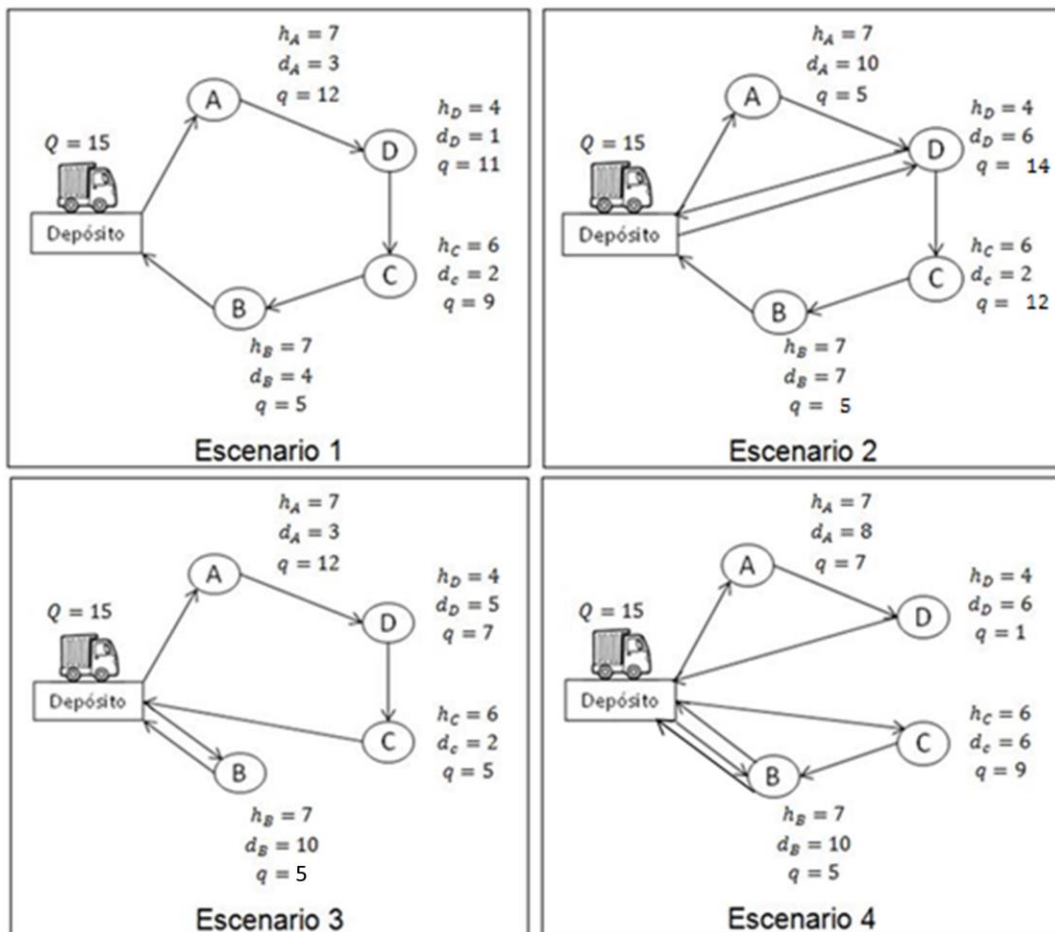
$$E2 = \{A = 10, D = 6, C = 2, B = 7\}$$

$$E3 = \{A = 3, D = 5, C = 2, B = 10\}$$

$$E4 = \{A = 8, D = 6, C = 6, B = 10\}$$

En la figura 11, se muestra el recorrido que debe realizar el vehículo para satisfacer la demanda de cada cliente en cada uno de los escenarios.

Figura 11. Escenarios para el cálculo de la función objetivo.



1. En el Escenario 1 se puede observar que el vehículo puede realizar su ruta sin ningún reabastecimiento preventivo ni fallo en la ruta; esto es debido a que la capacidad residual (q) del vehículo siempre fue mayor que el valor umbral (h) de los distintos clientes; de igual forma, la capacidad residual (q) del vehículo fue mayor que la demanda real de cada cliente (d). El costo total de este escenario es igual a 11,3137 unidades.
2. En el escenario 2 se presenta un fallo en la ruta debido a que la capacidad residual (q) del vehículo sí es mayor que el umbral del cliente D (h_D), pero es menor que la demanda real de dicho cliente (d_D). En este caso el vehículo debe ir del cliente D al depósito y volver al cliente D a satisfacer su demanda. El costo total de este escenario es igual a 16,9705 unidades.
3. En el escenario 3 se presenta un reabastecimiento preventivo debido a que la capacidad residual del vehículo (q) es menor que el umbral del cliente B (h_B), por lo que el vehículo debe ir al depósito antes de ir al nodo B. El costo total de este escenario es 19,799 unidades.
4. En el escenario 4 se presenta tanto un fallo en la ruta como un reabastecimiento preventivo. El reabastecimiento preventivo sucede debido a que la capacidad residual del vehículo (q) es menor que el umbral del cliente C (h_C); el fallo en la ruta se presenta en el cliente B, por lo que el vehículo debe ir al depósito y volver al nodo B a satisfacer la demanda del cliente. El costo total de este escenario es de 28,2843 unidades.

Para calcular el costo total del viaje con esa ruta a priori, se realiza el cálculo del promedio de los cuatro costos presentados en cada uno de los distintos escenarios. Es así como el costo total es:

$$C_t = \frac{11,3136 + 16,9705 + 19,799 + 28,2843}{4} = 19,0919 \text{ unidades}$$

8. DEFINICIÓN DEL NÚMERO DE ESCENARIOS

Se debe recordar que el costo total esperado de una ruta específica está determinado, en gran medida, por el número de escenarios a utilizar. Esto es debido a la existencia de la variabilidad entre cada escenario presente. Es por esto que a menor número de escenarios, el costo se verá mayormente afectado por los valores extremos que se presenten; sin embargo, el tiempo computacional será insignificante; En cambio, a mayor número de escenarios la variabilidad será menor pero el tiempo computacional será mucho más significativo, por lo que se hace necesaria la realización de un análisis que logre determinar la cantidad de escenarios adecuada para obtener un valor del costo total esperado que sea aceptable en cuanto a estabilidad y tiempo computacional.

A partir de esto se realiza, en primera instancia, un estudio gráfico en el cual se observa la variación del valor del costo total esperado a medida que se aumenta el número de escenarios. Esta representación gráfica se realiza tanto para la ruta de 100 clientes como para la ruta de 200 clientes. Posterior a esto, se analizan las diferencias en tiempo y costos de los distintos números de escenarios seleccionados y a partir de esto se determina el número total adecuado a utilizar.

Figura 12. Costos vs número de escenarios para 100 clientes

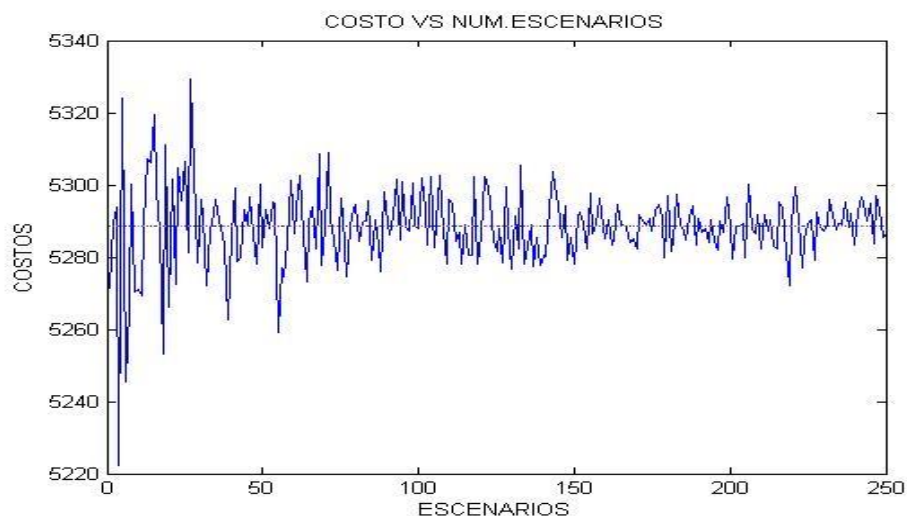
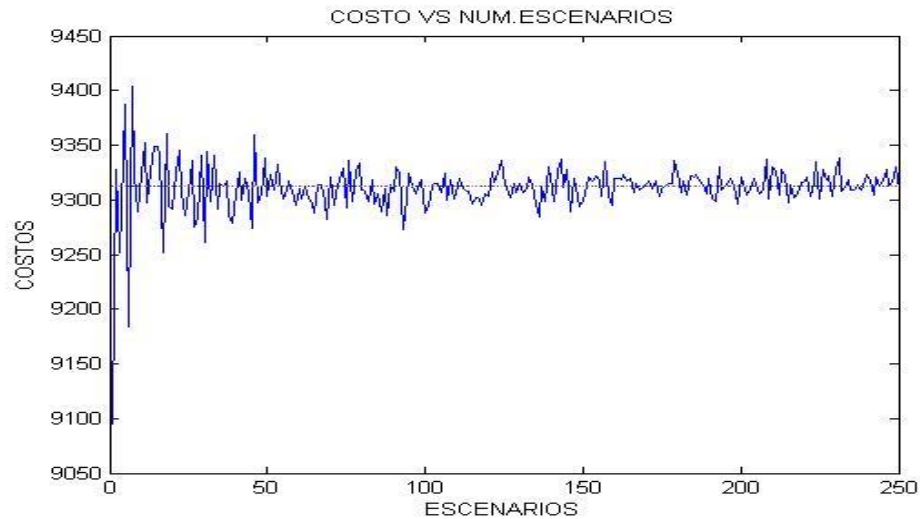


Figura 13. Costos vs número de escenarios para 200 clientes



De acuerdo a las figuras 12 y 13, se observa que a medida que se aumenta el número de escenarios, la variabilidad de los costos totales esperados disminuye. De esta manera, con base en el comportamiento decreciente de la variabilidad del costo y la información presentada en otras investigaciones sobre el tema¹⁰⁷, se estudiarán los costos y tiempos obtenidos para 50 y 100 escenarios, a fin de establecer el número de escenarios adecuado en términos de eficiencia y eficacia con el cual se desarrollará el algoritmo. En este sentido, se analizará la diferencia porcentual existente entre el menor valor obtenido y cada uno de los valores conseguidos en diferentes pruebas, con el objetivo de determinar el porcentaje de variabilidad existente con cada número de escenarios.

A continuación se presentan las tablas con los valores mínimos y máximos, obtenidos en costo con su respectivo tiempo computacional, para 50 y 100 escenarios.

¹⁰⁷ ACOSTA NIÑO, Francia del Pilar; OSORIO OSORIO, Daniela Angélica. Op. Cit.

Tabla 5. Costo y tiempo computacional para un número de escenarios 50 y 100 con una cantidad de 100 clientes.

NÚMERO DE ESCENARIOS PARA 100 CLIENTES						
50			100			
	VALOR	TIEMPO	DIF. PORCENTUAL	VALOR	TIEMPO	DIF. PORCENTUAL
Mínimo	5260,42939	9,37044905	0	5279,268	9,88717739	0
Máximo	5323,98125	10,0982589	1,208111646	5306,68889	10,1195934	0,519407046

Tabla 6. Costo y tiempo computacional para un número de escenarios 50 y 100 con una cantidad de 200 clientes.

NÚMERO DE ESCENARIOS PARA 200 CLIENTES						
50			100			
	VALOR	TIEMPO	DIF. PORCENTUAL	VALOR	TIEMPO	DIF. PORCENTUAL
Mínimo	9276,10627	9,28801967	0	9293,85206	9,638161	0
Máximo	9352,10188	10,097944	0,819261969	9339,96246	10,85969502	0,496138789

Se puede observar cómo, para 100 clientes, la diferencia porcentual de los valores con 50 escenarios es 2,3259 más grande que la diferencia porcentual presentada con 100 escenarios, mientras que el tiempo computacional es relativamente el mismo. Cuando se habla de 200 clientes pasa algo muy parecido, la diferencia porcentual con 50 escenarios es 1,6513 veces más grande que la diferencia porcentual con 100 escenarios y la diferencia en el tiempo computacional es pequeña. De esta forma, se establece trabajar con un número de 100 escenarios.

9. INSTANCIAS

Se denominan instancias a cada uno de los conjuntos de valores específicos que toman los parámetros del problema para ser evaluados. Estas instancias sirven para valorar la calidad de los distintos métodos de solución utilizados para resolver un mismo problema.

En el caso del VRPSD, los parámetros asociados al problema son:

- Número de los clientes
- Ubicación de los clientes
- Demanda de los clientes
- Capacidad del vehículo
- Política de servicio

Las instancias a utilizar en esta investigación son las desarrolladas en el trabajo “Optimización por simulación basado en un sistema evolutivo de optimización de enjambre de partículas para el problema de ruteo de vehículos con demandas estocásticas” realizado por la ingeniera Silvia Adriana Galván Núñez¹⁰⁸, con el fin de confrontar los resultados alcanzados y verificar la efectividad del algoritmo.

La elaboración de las distintas instancias se basa en dos niveles (1 y 2) para cada uno de los parámetros que se contemplan en el problema. Es así como el número de clientes tiene un nivel bajo o nivel 1 en el cual se asumen 100 clientes y un nivel alto o nivel 2 con 200 clientes. La ubicación de los clientes en el nivel 1 sigue una distribución uniforme en el intervalo de [0,100] mientras en el nivel 2, los clientes están repartidos aleatoriamente siguiendo una distribución normal con dos centros o dos clusters. La desviación y media para cada cluster se determinó de acuerdo a una distribución uniforme en el intervalo [0,100].

¹⁰⁸GALVÁN NÚÑEZ, Silvia Adriana; ARIAS, Javier; LAMOS, Henry. Optimización por simulación basado en un sistema evolutivo de optimización de enjambre de partículas para el problema de ruteo de vehículos con demandas estocásticas. Tesis de maestría de ingeniería industrial. Bucaramanga. Universidad Industrial de Santander. 2012.

La demanda de cada uno de los clientes tanto en el nivel 1 como en el nivel 2 sigue una distribución de probabilidad uniforme discreta, guiada por los valores de la demanda promedio de cada cliente (D_i) y la desviación (S_i) del mismo. De forma que la demanda real para el cliente i toma valores que se encuentren en el intervalo $[D_i - S_i, D_i + S_i]$. El valor umbral utilizado para realizar el reabastecimiento preventivo para el nivel 1 está dado por la demanda promedio de cada cliente y en el nivel 2 está dado por el primer cuartil del intervalo de demandas para cada cliente.

La capacidad del vehículo se calcula mediante la siguiente ecuación¹⁰⁹:

$$Q = \frac{\sum_{i=1}^n D_i * dp}{n} \quad [28]$$

Donde:

dp es el número promedio de clientes atendidos por el vehículo antes de realizar la descarga preventiva. Para el nivel 1 toma un valor de 4 y para el nivel 2 un valor de 8.

Tabla 7. Factores y niveles.

Factores	Nivel 1	Nivel 2
Número de clientes (n)	100	200
Ubicación de los clientes (P)	Distribución uniforme	Distribución normal con dos clusters
Demanda promedio (D_i)	[50,70]	[80,100]
Desviación (S_i)	[1,5]	[10,50]
Número de clientes atendidos antes de la descarga preventiva (dp)	4	8
Umbral (h)	Demanda promedio	Primer cuartil de la demanda

¹⁰⁹ GENDREAU, Michel., et al. An Exact Algorithm for the Vehicle Routing Problem with Stochastic Demands and Customers. En: Transportation Science. 1995. Vol. 29, no. 2, p. 143–155.

En la tabla 8, se muestran las ocho instancias planteadas por Silvia Galván como combinaciones de los distintos niveles y factores.

Tabla 8. Instancias

Instancias	<i>n</i>	<i>P</i>	<i>D_i</i>	<i>S_i</i>	<i>r</i>	<i>h</i>
1	Nivel 1	Nivel 1	Nivel 1	Nivel 1	Nivel 1	Nivel 1
2	Nivel 1	Nivel 1	Nivel 2	Nivel 2	Nivel 2	Nivel 2
3	Nivel 1	Nivel 2	Nivel 1	Nivel 2	Nivel 2	Nivel 1
4	Nivel 1	Nivel 2	Nivel 2	Nivel 1	Nivel 1	Nivel 2
5	Nivel 2	Nivel 1	Nivel 1	Nivel 2	Nivel 1	Nivel 2
6	Nivel 2	Nivel 1	Nivel 2	Nivel 1	Nivel 2	Nivel 1
7	Nivel 2	Nivel 2	Nivel 1	Nivel 1	Nivel 2	Nivel 2
8	Nivel 2	Nivel 2	Nivel 2	Nivel 2	Nivel 1	Nivel 1

10. DESARROLLO DEL ALGORITMO PARA EL VRPSD

Las etapas de la metaheurística Optimización por espiral para dar solución al problema de ruteo de vehículos con demandas estocásticas (VRPSD) se detallan a continuación.

Para la primera iteración:

Etapa 1. Inicialización de los parámetros y factores.

Etapa 2. Cálculo de las distancias euclidianas entre cada cliente.

Etapa 3. Inicialización de las espirales mediante la heurística del vecino más cercano.

Para las demás iteraciones:

Etapa 4. Se establecen las demandas de cada cliente para cada uno de los escenarios.

Etapa 5. Cálculo del costo de cada una de las espirales.

Etapa 6. Se establece como x^* a la espiral que haya obtenido el menor costo o menor valor de la función objetivo.

Etapa 7. Mejoramiento de la ruta aplicando la heurística OPT y operadores de mutación.

Etapa 8. Para cada espiral, se decodifica la ruta de clientes.

Etapa 9. Se actualiza la posición de cada una de las espirales.

Etapa 10. Se realiza la codificación de cada una de las espirales, obteniendo la ruta de clientes actualizada.

Etapa 11. Si aún no se ha cumplido el criterio de parada, ir a la etapa 4. En caso contrario, la solución del VRPSD es el costo de la ruta x^* obtenido en la última iteración.

1. Inicialización de los parámetros.

En esta etapa, se establecen los parámetros propios de la metaheurística optimización por espiral, los cuales son: el radio de convergencia (r), el ángulo de rotación (θ) y el número de espirales a utilizar (m).

Así mismo, en esta etapa se determina la instancia o experimento a evaluar y junto con esto, se establecen todos los parámetros pertenecientes al experimento; estos son el número de clientes, la posición de los clientes ($x_j - y_j$), la demanda promedio (D_j) y la desviación (S_j) de cada cliente, el valor umbral para cada cliente (h_j), el número de clientes promedio antes de realizar un reabastecimiento preventivo (dp) y la capacidad (Q) que tendrá el vehículo.

2. Cálculo distancias euclidianas.

Dado que ya se cuenta con la posición ($x_j - y_j$) de cada uno de los clientes, se procede a realizar una matriz cuadrada (*clientesxclientes*) que contenga las distancias euclidianas entre cada uno de estos, con el fin de facilitar posteriormente el cálculo del costo de cada ruta.

3. Inicialización de las espirales.

Para la inicialización de las espirales, las cuales contienen las rutas que realiza el vehículo, se procedió primero a analizar la influencia de realizar la etapa de inicio de manera aleatoria y mediante la heurística el vecino más cercano, esto con el fin de determinar si la manera de constituir las espirales afecta considerablemente la metaheurística.

En la tabla 9, se puede observar los resultados obtenidos cuando se aplicó la metaheurística optimización por espiral iniciando las espirales con la heurística el vecino más cercano y los resultados cuando se iniciaron las espirales de manera aleatoria; es así como se confirma que cuando se inicia con la heurística, el costo

de viaje se logra obtener mejores resultados que cuando se inicia de forma aleatoria. La mayor diferencia porcentual se presenta en la instancia 2 con un valor de 120,15% mientras que la instancia 6 con un valor de 18,92% es quien presenta la menor diferencia porcentual.

Tabla 9. Comparación en la inicialización de las espirales.

Vecino más cercano- Optimización por espiral							
Instancia 1	Instancia 2	Instancia 3	Instancia 4	Instancia 5	Instancia 6	Instancia 7	Instancia 8
5265,42585	3001,18234	2390,99594	4147,98838	9271,26331	5211,04019	4165,09761	8019,60648
Aleatorio- Optimización por espiral							
Instancia 1	Instancia 2	Instancia 3	Instancia 4	Instancia 5	Instancia 6	Instancia 7	Instancia 8
7897,42154	6607,05281	4459,34177	6131,56684	15904,8272	6197,21079	9143,32514	12348,8107

Dado estos resultados, se opta por realizar la inicialización de las espirales mediante la heurística el vecino más cercano.

4. Demandas clientes por escenarios.

Teniendo definidas las rutas que realizará el vehículo y sabiendo que con un número de escenarios de 100 la función objetivo tiende a estabilizarse; es necesario establecer el conjunto de demandas que presentará cada cliente en los distintos escenarios de acuerdo a su demanda promedio (D_j) y a su desviación (S_j). Esto se realiza mediante la elección de 100 valores presentes en el intervalo $D_j \pm S_j$, donde todos los valores tienen la misma probabilidad de ser seleccionados.

5. Costo de espirales.

Para cada uno de los escenarios se realiza el cálculo del costo teniendo en cuenta si se presenta reabastecimiento preventivo o fallos en la ruta. El valor de cada

escenario es almacenado para posteriormente calcular el valor promedio del costo para cada ruta o para espiral.

6. Establecimiento del valor x^*

Calculado el costo promedio para cada una de las rutas, se debe seleccionar la que presente el menor costo y ésta se define como el valor x^* , el cual será el punto al cual van a converger todas las espirales en la iteración o centro de rotación para todas las espirales presentes.

7. Mejoramiento de la ruta.

Para el mejoramiento de la ruta, se utilizarán los operadores de mutación y la heurística 2-Opt.

7.1 Operador de mutación

En primera instancia, se aplica el operador mutación con el fin de realizar una mayor exploración en el espacio de búsqueda y garantizar una mejor solución al problema.

Para este caso, la mutación se realizará a las m espirales utilizadas en la primera mitad de las iteraciones planteadas. Existirán tres maneras de realizar la mutación y éstas se seleccionarán de forma aleatoria para cada espiral. Posterior a este proceso, se elegirán las m espirales (tanto mutadas como no mutadas) que tengan el mejor valor en la función objetivo.

Para la realización de las tres mutaciones posibles, el primer paso es la selección de dos valores aleatorios que serán los puntos de inicio y fin de la mutación, punto I será el menor entre los dos y será el inicio de la mutación, el punto J será el mayor y fin de la mutación. Posterior a esto, se elegirá la mutación a realizar entre mutación flip (vuelta), mutación swap (intercambio) y mutación slide (deslizamiento).

La mutación se realiza a la ruta de cada una de las m espirales. A continuación, se explica de forma gráfica cada una de las posibles mutaciones.

- **Mutación Flip.** Se invierte el orden en la ruta, teniendo en cuenta el inicio de la mutación (I) y el fin de la misma (J).

Figura 14. Mutación Flip.



- **Mutación Swap.** En este caso, sólo se intercambian los valores que se encuentran en las posiciones I y J .

Figura 15. Mutación Swap.



- **Mutación Slide.** Para esta mutación, el valor perteneciente al nodo $I + 1$ se ubica en la posición I , el valor de $I + 2$ en la posición $I + 1$ y así sucesivamente; el valor ubicado en la posición I se ubicará en la posición J .

Figura 16. Mutación Slide.



7.2 Heurística 2-Opt

En segunda instancia, se aplica la heurística 2-Opt con el fin de mejorar la ruta que tiene el menor costo. Es así como, a partir de la ruta x , contenida en la menor espiral, se eliminan 2 arcos no adyacentes y se reconectan los 2 nodos libres, de forma que se crea una nueva ruta x' . Ahora, si el costo de esta nueva ruta es menor que el costo de la ruta inicial, se reemplaza la ruta x por la ruta x' .

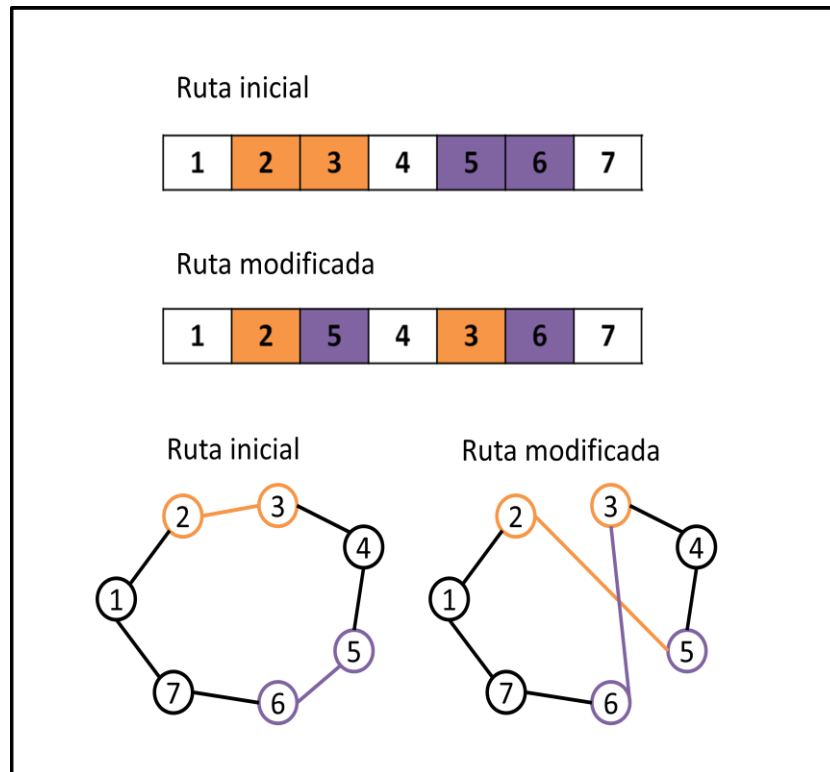
Es importante resaltar, que debido a la gran cantidad de posibles combinaciones el tiempo computacional requerido es bastante considerable, por lo cual se implementaron las siguientes estrategias a fin de mitigar este efecto:

- La heurística se detendrá en el momento que encuentre una ruta con menor costo que la inicial.
- Si en iteraciones anteriores, ya se han evaluado todas las posibles combinaciones de la ruta a evaluar, no se aplica la heurística.

- La heurística 2-Opt se implementa solamente a la primera mitad de las iteraciones planteadas.

En la siguiente imagen se puede observar un ejemplo del proceso que se realiza a la hora de aplicar el 2-Opt en el proceso.

Figura 17. Aplicación de 2-Opt.



8. Decodificación de la ruta.

Teniendo presente que las espirales contienen las rutas establecidas para que el vehículo visite a los clientes, éstas deben estar compuestas por número enteros que determinen los arcos i, j de la ruta que debe realizar el vehículo. Para la aplicación de la metaheurística optimización por espiral, es necesario convertir las rutas de las espirales de forma discreta a forma continua de manera que esas rutas queden en un espacio de solución factible para la aplicación de la rotación.

Para realizar dicha decodificación, se opta por la aplicación de la ecuación 27, resaltando que los valores de frontera a utilizar para la decodificación ($x_{min} = 0,1$ y $x_{max} = 0,7$), son tomados de la información ofrecida por Geetha, et al¹¹⁰.

9. Actualización de la posición de las espirales.

Teniendo las rutas en el espacio de solución de la metaheurística, se puede proceder a la actualización de las mismas, lo que implica la rotación de cada una de las espirales con centro en la espiral que obtuvo el menor costo, es decir, la espiral que se ha denominado como x^* .

A partir de la ecuación 19, cada espiral se rotará para obtener $m - 1$ nuevas rutas que deberán evaluarse, esto dado a que la espiral x^* quedará con los mismos valores puesto que está rotando sobre ella misma.

10. Codificación de la ruta.

Para la evaluación de las nuevas rutas obtenidas a partir de la rotación, es necesario volver a codificar las rutas de forma que pasen del espacio de solución de la metaheurística al espacio discreto que define los arcos i, j .

Es decir, se deben convertir de nuevo los valores continuos de posición del vector X_i en valores discretos del vector solución Y_i . Para esto se utiliza la regla ROV (Rank Order Value), creada por Bo, Wang y Jing¹¹¹ en la cual se asigna la valoración de 1 al menor valor dentro de cada espiral (Smallest Position Value, SPV). Posteriormente se asigna la valoración de 2 al segundo menor valor posición y así sucesivamente para los demás valores de la espiral.

¹¹⁰ Ibid., p.185

¹¹¹ LIU, Bo, WANG, Ling y JIN, Yi-hui. An effective hybrid PSO-based algorithm for flow shop scheduling with limited buffers. En: Computers & Operations Research. 2008. Vol. 35, p. 2791–2806. DOI 10.1016/j.cor.2006.12.013.

11. DISEÑO EXPERIMENTAL

11.1 DEFINICIÓN DE LOS PARÁMETROS DEL ALGORITMO OPTIMIZACIÓN POR ESPIRAL

El desempeño del algoritmo de la metaheurística optimización por espiral depende de los siguientes factores:

- El ángulo de rotación (θ)
- Radio de convergencia (r)
- Número de espirales (m)

Los niveles del número de espirales a utilizar son establecidos de acuerdo a la información presentada por Rafael Ortíz y Edgar García.¹¹² Por otro lado, para establecer los niveles del ángulo de rotación y el radio de convergencia, se realizaron 18 combinaciones de estos dos parámetros y se ejecutaron 5 veces cada una de estas combinaciones, para las instancias 1 y 7. En la tabla 10, se presentan las combinaciones realizadas. Posteriormente, se seleccionaron los valores de θ ($\pi/9$ y $\pi/4$) y r (0,9 y 0,99), los cuales obtuvieron los menores costos.

¹¹² ORTÍZ GELVES, Rafael Armando; GARCÍA MORANTES, Edgar Sneyder. Diseño de un absorbedor electromagnético multicapa mediante el método de la espiral. Tesis de pregrado de ingeniería electrónica. Bucaramanga. Universidad Industrial de Santander. 2013.

Tabla 10. Combinaciones de prueba

COMBINACIÓN	ANGULO	RADIO	COMBINACIÓN	ANGULO	RADIO
1	$\pi/9$	0,9	10	$\pi/3$	0,95
2	$\pi/4,5$	0,9	11	$\pi/2,25$	0,95
3	$\pi/4$	0,9	12	$\pi/2$	0,95
4	$\pi/3$	0,9	13	$\pi/9$	0,99
5	$\pi/2,25$	0,9	14	$\pi/4,5$	0,99
6	$\pi/2$	0,9	15	$\pi/4$	0,99
7	$\pi/9$	0,95	16	$\pi/3$	0,99
8	$\pi/4,5$	0,95	17	$\pi/2,25$	0,99
9	$\pi/4$	0,95	18	$\pi/2$	0,99

Tabla 11. Configuración de los valores de los niveles de cada factor

Factor	Nivel Bajo	Nivel Alto
Ángulo de rotación (Θ)	$\pi/9$	$\pi/4$
Radio de convergencia (r)	0,9	0,99
Número de espirales	20	30

11.2 DEFINICIÓN DEL DISEÑO EXPERIMENTAL

Teniendo los parámetros establecidos, se establece realizar un diseño de experimentos factorial 2^3 . Los 8 tratamientos a evaluar, fueron obtenidos usando el software estadístico Statgraphics Centurion y se presentan en la tabla 12.

Tabla 12.Tratamientos a utilizar en el diseño experimental

Ángulo	Radio de convergencia	Número de espirales
-1	-1	-1
1	-1	-1
-1	1	-1
1	1	-1
-1	-1	1
1	-1	1
-1	1	1
1	1	1

11.3 RESULTADO DEL DISEÑO EXPERIMENTAL

Es importante resaltar que cada tratamiento fue replicado 5 veces para el cálculo del valor promedio de la función objetivo. A continuación se presentan las características del equipo de cómputo mediante el cual se ejecutó el programa y los respectivos resultados obtenidos.

Tabla 13.Características del equipo de cómputo.

PROCESADOR	Intel Core i5 – 3.2[GHz]
MEMORIA RAM	8 GB
SISTEMA OPERATIVO	Windows 7- 64 bits
VERSIÓN MATLAB	Matlab® 7.14.0.739

Tabla 14. Resultado del diseño experimental.

Corridas			Valor promedio de la función objetivo							
A	B	C	Instancia 1	Instancia 2	Instancia 3	Instancia 4	Instancia 5	Instancia 6	Instancia 7	Instancia 8
-1	-1	-1	4933,31	2732,01	2179,21	3962,07	9188,96	4886,38	4055,14	8001,23
1	-1	-1	4912,22	2736,16	2160,71	3899,83	9179,73	4888,07	4050,51	8018,32
-1	1	-1	4927,90	2745,35	2189,44	3943,13	9211,07	4910,64	4041,71	8016,99
1	1	-1	4908,14	2731,84	2177,07	3942,00	9184,75	4918,17	4060,08	8021,95
-1	-1	1	4940,37	2722,06	2147,42	3949,49	9190,24	4882,71	4055,11	8014,13
1	-1	1	4926,44	2747,30	2160,88	3941,97	9197,39	4895,83	4044,66	8013,23
-1	1	1	4949,48	2750,25	2161,65	3944,16	9196,99	4906,07	4032,35	8017,18
1	1	1	4901,89	2773,50	2162,14	3937,99	9209,01	4905,41	4048,20	8013,15

En la tabla 14 se presentan los resultados obtenidos para cada instancia con cada uno de los tratamientos planteados. Es así como para cada una de éstas, existe una combinación de parámetros que logra ofrecer un mínimo costo; éste se encuentra resaltado en color rojo.

11.4 ANÁLISIS DEL DISEÑO EXPERIMENTAL

El modelo de regresión correspondiente al diseño de experimentos se expresa a continuación:

$$y = \beta_0 + \beta_1x_1 + \beta_2x_2 + \beta_3x_3 + \varepsilon$$

Donde:

y Es el costo total esperado.

β_0 Es el efecto medio global.

β_1x_1 Es el efecto producido por el parámetro A, es decir el ángulo de rotación.

β_2x_2 Es el efecto producido por el parámetro B, es decir el radio de convergencia.

β_3x_3 Es el efecto producido por el parámetro C, es decir el número de espirales.

ε Es el error aleatorio.

Teniendo los resultados del diseño experimental, en el cual se realizaron ocho tratamientos con cinco réplicas cada uno, se logró identificar los efectos de cada uno de los parámetros sobre la función objetivo de cada una de las instancias.

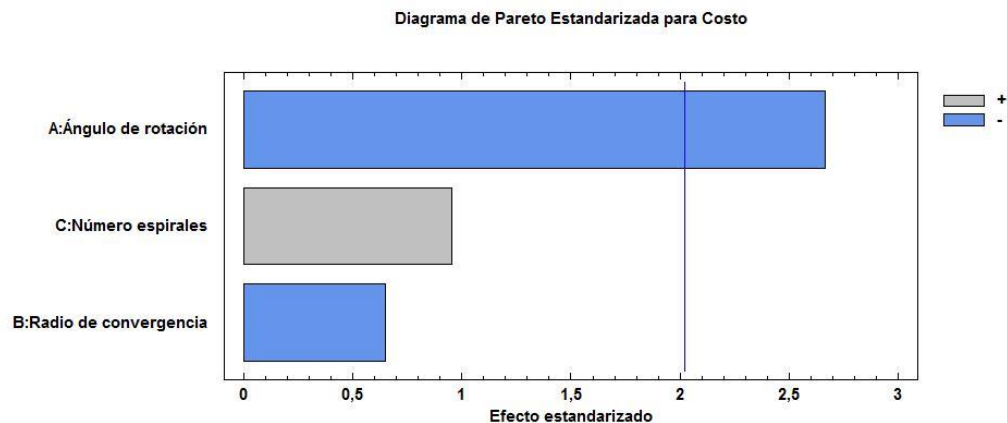
Tabla 15. Efectos estimados para la función objetivo.

Factores	Efecto estimado							
	Instancia 1	Instancia 2	Instancia 3	Instancia 4	Instancia 5	Instancia 6	Instancia 7	Instancia 8
A: Ángulo de rotación	-25,5937	2,49667	-4,73917	-18,8162	-6,89917	0,467917	6,97625	3,315
B: Radio de convergencia	-6,23125	23,1425	11,0275	3,03792	14,1733	26,7788	-7,95792	6,55667
C: Espirales	9,15542	4,64917	-19,0917	7,09042	4,47583	-8,26292	-4,59125	-1,16583

A partir de esto, se debe tener presente que si el valor del efecto es positivo significa que al pasar del nivel bajo al nivel alto, el valor de la función objetivo aumenta, lo que quiere decir que el costo total del viaje es mayor. Es así como, dependiendo de la instancia a estudiar, se establece qué factores deben estar en nivel bajo y qué factores en nivel alto.

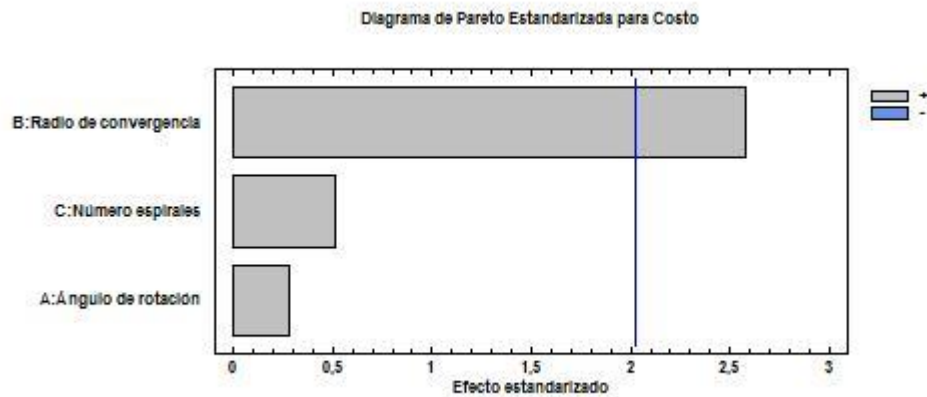
Así mismo, a continuación se presentan las gráficas de Pareto estandarizadas con el objetivo de identificar los parámetros de mayor influencia sobre la función objetivo.

Figura 18. Diagrama de Pareto para la instancia 1.



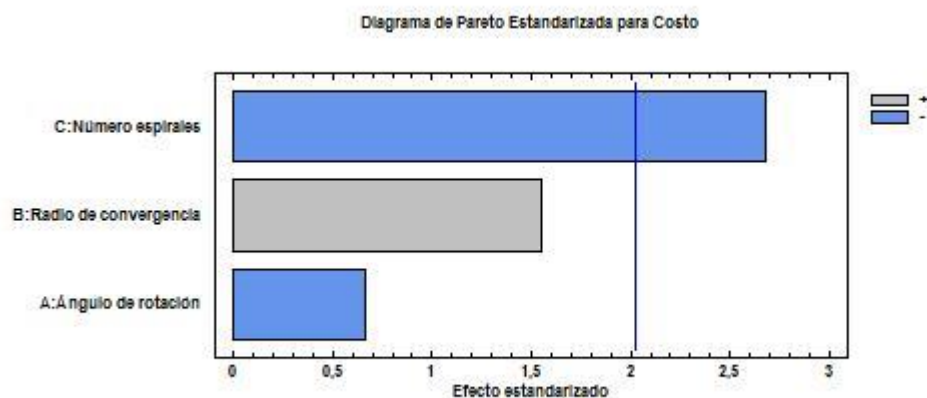
Como se puede observar en la figura 18, el ángulo de rotación ejerce un efecto significativo sobre la función objetivo; al ser este efecto negativo, se debe asumir el valor de dicho parámetro en nivel alto.

Figura 19. Diagrama de Pareto para la instancia 2.



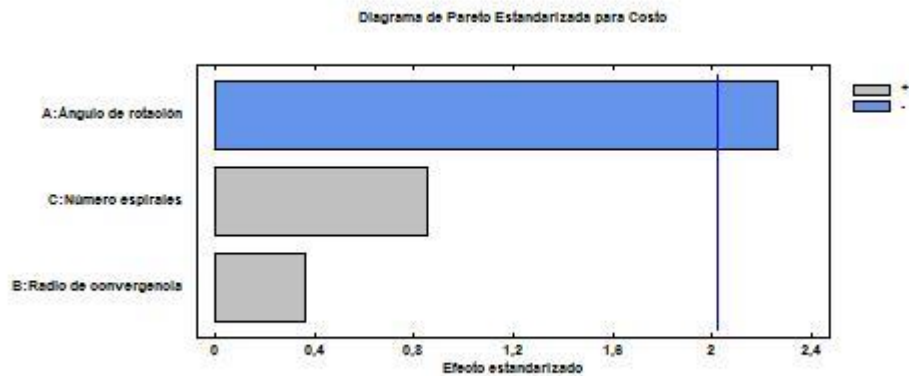
Para la instancia 2 se tiene que el parámetro que ejerce un efecto significativo es el radio de convergencia, el cual genera una influencia positiva lo que indica que debe ubicarse en el nivel bajo.

Figura 20. Diagrama de Pareto para la instancia 3.



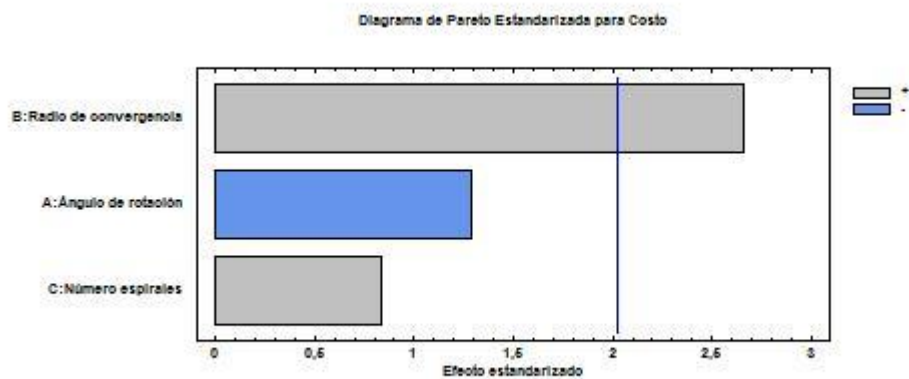
Mediante la información ofrecida por la figura 20, se tiene que el factor que ejerce un efecto significativo sobre la función objetivo es el número de espirales, el cual se debe ubicar en el nivel alto debido a su efecto negativo.

Figura 21. Diagrama de Pareto para la instancia 4.



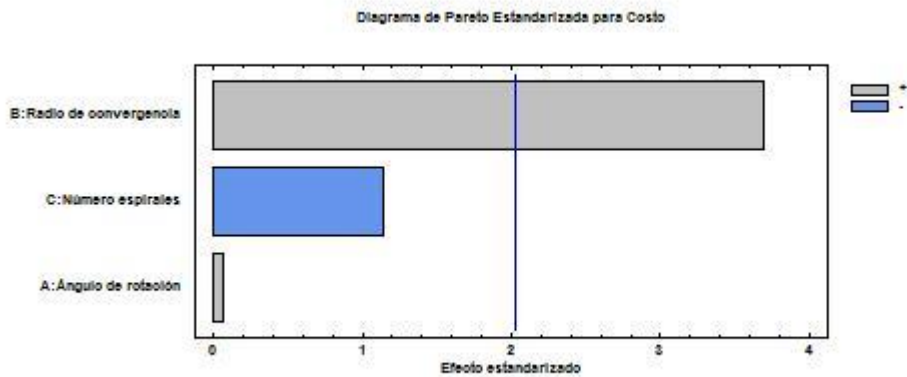
Para la instancia 4 se tiene que el ángulo de rotación es el único parámetro que ejerce una influencia significativa. En este caso, éste debe ubicarse en el nivel alto debido a su efecto negativo.

Figura 22. Diagrama de Pareto para la instancia 5.



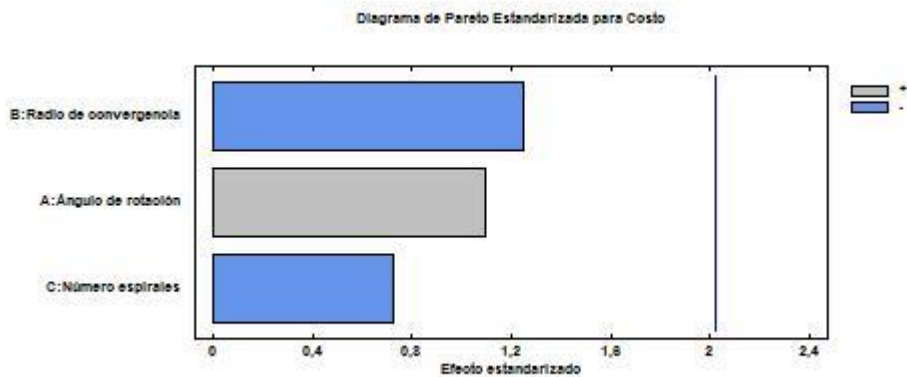
Como se puede observar en la figura 22, el radio de convergencia es el único parámetro que ejerce un efecto significativo y éste es positivo, lo cual indica que el nivel a utilizar debe ser el bajo.

Figura 23. Diagrama de Pareto para la instancia 6.



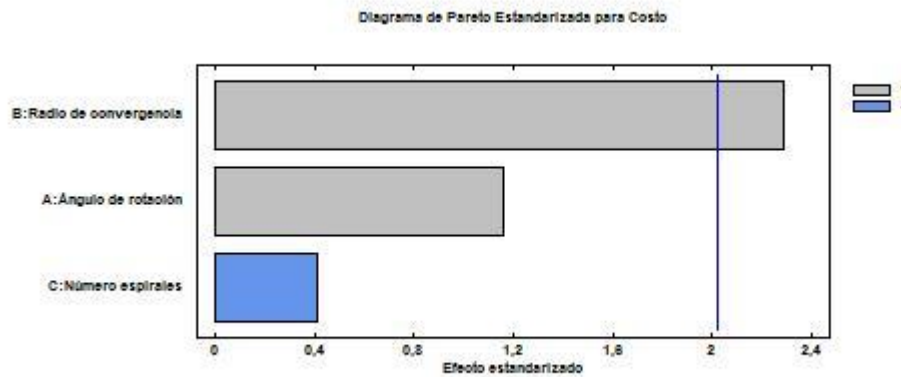
En la instancia 6, al igual que en la instancia 6, el único factor que genera una influencia significativa es el radio de convergenola. El nivel a utilizar debe ser el bajo debido a que su efecto es positivo.

Figura 24. Diagrama de Pareto para la instancia 7.



Para la instancia 7, mediante la figura 24 se puede determinar que no existe ningún parámetro que genere un efecto significativo sobre la función objetivo.

Figura 25. Diagrama de Pareto para la instancia 8.



Por medio de la figura 25, se logra determinar que el único factor que genera una influencia significativa sobre la función objetivo es el radio de convergenola. Para este caso, el efecto generado es positivo lo que indica que dicho parámetro debe ubicarse en el nivel bajo.

11.5 COMPARACIÓN DEL RENDIMIENTO DEL ALGORITMO

En primera instancia, se realiza la comparación de los resultados del algoritmo optimización por espiral básico con los obtenidos agregando los operadores de mutación y la heurística 2-Opt. Los resultados se presentan en la tabla 16.

Tabla 16. Resultados del valor de la función objetivo con Optimización por Espiral e Híbrido de Optimización por Espiral

INSTANCIA	ESPIRAL MODIFICADO	ESPIRAL BÁSICO	DIFERENCIA PORCENTUAL
1	4901,89	5248,86	7,08%
2	2722,06	2979,95	9,47%
3	2147,42	2352,43	9,55%
4	3899,83	4125,40	5,78%
5	9179,73	9260,12	0,88%
6	4882,71	5202,06	6,54%
7	4032,35	4160,70	3,18%
8	8001,23	8020,52	0,24%

Con base en los anteriores resultados, se evidencia el efecto positivo tanto de los operadores de mutación como de la heurística 2-Opt para obtener mejores resultados ya que en todas las instancias comparadas se obtuvieron menores costos respecto de la metaheurística básica. Esto se debe esencialmente, a que la heurística 2-Opt aumenta la intensificación de la búsqueda alrededor de buenas soluciones encontradas y los operadores de mutación favorecen la diversificación, es decir, que buscan mejores soluciones al abarcar una mayor región.

En segunda instancia, se realiza la comparación de los resultados presentados en el artículo “Optimización por simulación basado en un sistema evolutivo de optimización de enjambre de partículas para el problema de ruteo de vehículos con demandas estocásticas” realizado por la ingeniera Silvia Adriana Galván Núñez¹¹³ y los obtenidos en la presente investigación.

¹¹³GALVÁN NÚÑEZ, Silvia Adriana; ARIAS, Javier; LAMOS, Henry. Optimización por simulación basado en un sistema evolutivo de optimización de enjambre de partículas para el problema de ruteo de vehículos con demandas estocásticas. Tesis de maestría de ingeniería industrial. Bucaramanga. Universidad Industrial de Santander. 2012.

Tabla 17. Resultados de la función objetivo con Híbrido de Optimización por espiral y EPSO.

INSTANCIA	OPTIMIZACIÓN POR ESPIRAL	EPSO	DIFERENCIA PORCENTUAL
1	4901,89	5103,46	4,11%
2	2722,06	2877,53	5,71%
3	2147,42	2257,82	5,14%
4	3899,83	4074,92	4,49%
5	9179,73	9274,82	1,04%
6	4882,71	5093,08	4,31%
7	4032,35	4128,65	2,39%
8	8001,23	8005,36	0,05%

A partir de los resultados presentados en la tabla 17, se observa que en todas las instancias evaluadas, el algoritmo desarrollado en la presente investigación supera los resultados obtenidos por el algoritmo híbrido EPSO, con mejoras de hasta el 5,71%.

12. CONCLUSIONES

Con base en los resultados obtenidos para las diferentes instancias y su respectiva comparación con otros métodos, se valida que el algoritmo optimización por espiral es una metaheurística viable y eficaz para la solución del problema de ruteo de vehículos con demandas estocásticas.

Se establece la importancia de la aplicación de la heurística 2-Opt como de los operadores de mutación para la obtención de mejores valores de la función objetivo, gracias a que estos logran mejorar las estrategias de búsqueda de intensificación y diversificación del algoritmo.

La metaheurística optimización por espiral presenta una serie de características que incentivan su elección para la solución de problemas prácticos como es la consecución de resultados factibles en tiempos computacionales pequeños y la dependencia a un menor número de parámetros (ángulo de rotación y radio de convergencia).

A partir de los datos obtenidos en el diseño de experimentos, se evidencia que no existe una única combinación óptima para todas las instancias sino que ésta depende de las características propias de la instancia que se esté estudiando.

En cuanto el efecto de los parámetros, se observa el radio de convergencia es el parámetro que más influye en el resultado obtenido; esto debido a que en las instancias 2,5, 6 y 8 es éste el único que genera un efecto significativo. Así mismo, el número de espirales es el parámetro con menor influencia, siendo significativo para una sola instancia.

13.RECOMENDACIONES

Para futuras investigaciones, se recomienda:

Establecer un nuevo banco de pruebas con el fin de seguir estudiando los efectos presentados tanto por las interacciones como por cada parámetro sobre el valor de la función objetivo; de forma que se pueda llegar a una conclusión más genérica sobre los niveles que debe tomar cada uno de estos elementos.

Aplicar la metaheurística optimización por espiral a problemas cuyo espacio de solución sea de forma continua, a fin de aprovechar en mayor medida las características de intensificación y diversificación propias de las espirales logarítmicas.

Solucionar problemáticas empresariales mediante el algoritmo desarrollado en la presente investigación; esto con el objetivo de agregar al análisis un estudio del tiempo computacional y así poder optimizar el número de iteraciones tanto del problema como de los algoritmos utilizados para mejorar las soluciones.

Aplicar la metaheurística optimización por espiral para diferentes problemas de la Ingeniería Industrial con el objetivo de seguir comprobando la efectividad de esta metaheurística y conseguir su reconocimiento dentro de la comunidad científica.

BIBLIOGRAFÍA

ACOSTA NIÑO, Francia del Pilar; OSORIO OSORIO, Daniela Angélica. Un método evolutivo de colonia de hormigas para la solución del problema de ruteo de vehículos con demandas estocásticas. Tesis de pregrado de Ingeniería Industrial. Bucaramanga. Universidad Industrial de Santander. 2013.

AI, The Jin; KACHJTYICHYANUKUL, Voratas. A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. En: Computers & Operations Research 36. 2009. 10 p.

AK, Aykagan; ERERA, Alan L. A Paired- Vehicle Recourse Strategy for the Vehicle-Routing Problem with Stochastic Demands. En: Transportation science. Mayo 2007, Vol. 41, p. 222-237.

ARCHETTI, C.; SPERANZA, M.G.; HERTZ, A. A tabu search algorithm for the Split delivery vehicle routing problema. En: Transportation science. Frebero 2006, Vol 40. No. 1., p. 64-73, ISSN 970-26-0540-7.

BALLOU, Ronald H. Logística. Administración de la cadena de suministro. 5° Edición. Prentice Hall. Enrique Quintanar Duarte. México, 2004, 808 p.

BENAVENTE, Magdalena; BUSTOS, Jaime. Estado del arte en el problema de ruteo de vehículos (VRP). En: Laboratorio de Ingeniería aplicada, departamento de ingeniería de sistemas, Universidad de La Frontera. 13 p.

BIANCHI, Leonora; et al. Metaheuristics for the Vehicle Routing Problem with Stochastic Demands. En: Parallel Problem Solving from Nature-P.SN. Enero, 2004. vol. 8. 450 p.

BIANCHI, Leonora, et al. Hybrid Metaheuristics for the Vehicle Routing Problem with Stochastic Demands. En: Journal of Mathematical Modelling and Algorithms. 2006, p. 91-110.

CHEPURI, Krishna;DE-MELLO, Tito. Solving the Vehicle Routing Problem with Stochastic Demands using the Cross-Entropy Method.En: Annals of Operations Research.Febrero 2006. Vol.134, No. 1, p. 153-181.

CORDEAU, J.-F.; DESAULNIERS, Guy; DESROSIERS, Jacques; et al. The VRP with Time Wondows. En: GERAD. Febrero 1999. Revisado Junio 2000. 38 p.

DAZA, Julio Mario; MONTOYA, Jairo R; NARDUCCI, Francesco. Escuela de Ingeniería de Antioquia, Medellín (Colombia). Resolución del problema de enrutamiento de vehículos con limitaciones de capacidad utilizando un procedimiento metaheurístico de dos fases. En: Revista EIA, Diciembre 2009, No 12, p. 23-38, ISSN 1794-1237.

DEDONG, Wang; QIJUM, Chen; Lili. Stochastic Vehicle Routing Problems and their Solution Algorithm. En: International Conference on Automation and Logistics. Qingdao, China. Septiembre 2008., p. 958-962.

DROR, M.; LAPORTE, G. y TRUDEAU, P. Vehicle routing with stochastic demands: Properties and solution frameworks. En: Transportation Science. Agosto 1989, Vol. 23, No. 3, p. 166.

FAN, Jianhua; et al. A Multiple Vehicles Routing Problem Algorithm with Stochastic Demand.En: 6th World Congress on Intelligent Control and Automation (2006, Dalian), p.1688-1692.

GALVÁN NÚÑEZ, Silvia Adriana; ARIAS, Javier; LAMOS, Henry. Optimización por simulación basado en un sistema evolutivo de optimización de enjambre de partículas para el problema de ruteo de vehículos con demandas estocásticas. Tesis de maestría de ingeniería industrial. Bucaramanga. Universidad Industrial de Santander. 2012.

GALVIS MEJÍA, Jessica Tatiana; JAIMES MONSALVE, Gladys Andrea; QUIROGA BERDUG, Nataly Jhoana. Estudio cuantitativo de tres aplicaciones diferentes del problema de ruteo de vehículos (vrp) en la universidad industrial de Santander. Tesis de pregrado de Ingeniería Industrial. Universidad Industrial de Santander. 2011.

GARCÍA SABATER, José Pedro; MAHEUT, Julien. Modelo y métodos de investigación de operaciones. Procedimientos para pensar. Modelado y resolución de problemas de organización industrial mediante programación matemática. En: Universitat Politècnica de València. Grupo de investigación ROGLE. Departamento de organización de empresas. Curso 2011/2012. 181 p.

GEETHA, S.; POONTHALIR, G.; VANATHI, P.T. Metaheuristic approach for the Multi-depot Vehicle Routing Problem. En: Applied Artificial Intelligence. 2012, Vol. 26, No. 9, p. 878-901.

GOODSON, Justin C.; OHLMANN, Jeffrey W.; THOMAS, Barrett W. Cyclic-order neighborhoods with application to the vehicle routing problem with stochastic demand. En: European Journal of Operational Research. 2012, Vol. 217, p. 312–323.

HILLIER, Frederick S.; LIEBERMAN, Gerald J. Introducción a la investigación de operaciones. 9° Ed. Ciudad de México, México: Mc Graw Hill. 2010, 978 p.

IGLESIAS MARTÍNEZ, José; y otros. Complejidad Computacional. Grado Ingeniería informática. Teoría de Autómatas y Lenguas formales. [En línea] Universidad Carlos III de Madrid. [Citado el 12 de Julio del 2014] Disponible en: <<http://ocw.uc3m.es/ingenieria-informatica/teoria-de-automatas-y-lenguajes-formales/material-de-clase-1/tema-8-complejidad-computacional>>

ISMAIL, Zuhaimy; IRHAMAH. Genetic Algorithm and Tabu Search for Vehicle Routing Problems with Stochastic Demand. En: International Conference on Mathematical Science. Instituto Americano de Física. 2010, p. 488 – 504.

JABALI, Ola; REI, Walter; GENDREAU, Michel; LAPORTE, Gilbert. Partial-route inequalities for the multi-vehicle routing problema with stochastic demands. En: ScienceDirect. Discrets Applied Mathematics. 2014, 16 p.

JUAN, Angel A, et al. Applying Simulation and Reliability to Vehicle Routing Problems with Stochastic Demands. En: XI Conference of the Italian Association for Artificial Intelligence (AIIA09), Reggio-Emilia (Italia). 2009. P. 2

LI, Hong-Gui; LI, Xing-Guo. Image segmentation with pseudo branch and bound algorithm. En: Proceedings of the Eighth International Conference on Machine Learning and Cybernetics, Baoding, 12-15 Julio de 2009, 5 p.

LIU, Bo, WANG, Ling y JIN, Yi-hui. An effective hybrid PSO-based algorithm for flow shop scheduling with limited buffers. En: Computers & Operations Research. 2008. Vol. 35, p. 2791–2806. DOI 10.1016/j.cor.2006.12.013.

MARINAKIS, Yannis; MARINAKI, Magdalene. Combinatorial Expanding Neighborhood Topology Particle Swarm optimization for the Vehicle Routing Problem with Stochastic Demands. En: GECCO'13. 6 de Julio de 2013, p. 49-56.

MARTÍ, Rafael. Procedimientos metaheurísticos en optimización combinatoria. En: Departament d'Estadística i Investigació Operativa. Facultat de Matemàtiques. Universitat de València. 60 p.

MARTÍNEZ GARCÍA, F. Javier; MORENO PÉREZ, José A. Optimización por enjambre para la p-mediana continua y discreta. En: Actas del U congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados. 2007, p. 53-60, ISBN 978-84-690-3470-5.

MINETTI, Gabriela F. Una solución de computación evolutiva para el TSP, su posible aplicación en las organizaciones. Capítulo 7: un algoritmo evolutivo avanzado para TSP. En: Universidad Nacional de la Plata. Facultad de Informática. Magister en automatización de Oficinas. Octubre de 2000, p. 84-92.

MOLINA, José; EGUÍA, Ignacio. PROBLEMA DE OPTIMIZACION DE RUTAS DE VEHICULOS CON ASPECTOS MEDIOAMBIENTALES. Tesis de maestría en Organización Industrial y Gestión de Empresas. Sevilla. Universidad de Sevilla. Noviembre 2013.

NAGY, Bábor; SALHI, Saïd. Heuristic Algorithms for single and multiple depot vehicle routing problems with Pickups and deliveries. En: Working paper series. No.42. Canterbury Business School. Noviembre 2003. 28 p.

NOVOA, Clara; STORER, Robert. An approximate dynamic programming approach for the vehicle routing problem with stochastic demands. En: European Journal of Operational Research. 2008, p. 509-512.

OLLER, José A. Mecánica Cuántica. En: Departamento de física. Universidad de Murcia. 251 p.

PARLAKTUNA, Osman; IPAHIOGLU, Aydin S.; YAZICI, Ahmet. A VRP-Based Route Planning for a Mobile Robot Group. En: Turk J Elec Engin. 2007, Vol.15, NO.2, 12p.

PENG, Yong; SHU, Hai-ying. Research on Vehicle Routing Problem with Stochastic Demand and PSO-DP Algorithm with Inver-over Operator. En: ScienceDirect. Systems Engineering- theory & Practice. Octubre 2008, Vol. 28, p. 76-81.

QI, Chengming; CUI, Shoumei; SUN, Yunchuan. A two-Stage Hybrid Ant Colony Algorithm for the CVRP.En. International Conference on Computational Intelligence and Security. 2009. 5p.

QI, Chengming; CUI, Shoumei; SUN, Yunchuan. Using ant colony system and local search methods to solve VRPTW. EN: Pacific-Asia Workshop on Computational Intelligence and Industrial Ap.lication. 2008, p. 478-482.

QI, Chengming; SUN, Yunchuan. An Improved Ant Colony Algorith for VRPTW. En: International Conference on Computer Science and Software Engineering. 2008, p. 455-458.

ROCHA, L.; GONZÁLEZ, C.; ORJUELA, J. Una revisión al estado del arte del problema de ruteo de vehículos: Evolución histórica y métodos de solución. En: Ingeniería. 2011, Vol. 16, No. 2, p. 35 - 55.

RODRIGUEZ PÉREZ, Jorge. Caracterización, modelado y determinación de las rutas de la flota en una empresa de Rendering.En: E-Reding. Trabajos y proyectos fin de estudios de la E.T.S.I. Máster en Organización Industrial y Gestión de Empresas. 2012, 700 p.

SECOMANDI, Nicola. Comparing neuro-dynamic programming algorithms for the vehicle routing problem with stochastic demands. En: Computers & Operations Research. 2000. Vol. 27, 1203 p.

SECOMANDI, Nicola. A rollout policy for the vehicle routing problem with stochastic demands. En: Operations Research. 2001. Vol. 49, no. 5, 802 p.

SECOMANDI, Nicola. Analysis of a Rollout Approach to Sequencing Problems with Stochastic Routing Applications. En: Journal of Heuristics. Septiembre 2003. Vol. 9, No. 4, p. 321-352.

SULIEMAN, Dalia; JOURDAN, Laetitia; TALBI, El-Ghazali. Using multiobjective metaheuristics to solve VRP with uncertain demands. En: Evolutionary Computation (CEC). 2010, 8 p.

STEWART, W y GOLDEN, B. Stochastic vehicle routing: A comprehensive approach. En: European Journal of Operational Research. Diciembre, 1983, Vol. 14, p. 371-385.

TAMURA, Kenichi; YASUDA, Keiichiro. Spiral Optimization. A new multipoint search method. En: Systems, Man, and Cybernetics (SMC). Octubre 2011. 6 p.

TAN, K; CHEONG, C; GOH, C. Solving multiobjective vehicle routing problem with stochastic demand via evolutionary computation. En: European Journal of Operational Research. Marzo 2007. Vol. 177, No. 2, p. 813-839.

TRIPATHI, Mukul; KURIGER, Glenn, WAN, Hung-da. An ant based simulation optimization for vehicle routing problema with stochastic demands. En: Winter Simulation Conference. 2009, p. 2476-2487.

Universidad de Buenos Aires. Complejidad – problemas NP-COMPLETEs. [En línea] Algoritmos y estructuras de datos III. [Citado el 12 de Julio del 2014] Disponible en: <http://www2.dc.uba.ar/materias/aed3/201101/Documents/algo3_compl.pdf>

YANG, Wen-Huei;MATHUR, Kamlesh;BALLOU, Ronald.Stochastic vehicle routing problem with restocking.En: Transportation Science Trans. Sci. 2000, Vol. 34, No.1, p. 99-112.

ANEXOS

ANEXO A. FUNCIONES DE PRUEBA

Para la reproducción de los resultados, se evaluarán cada una de las funciones mediante la aplicación de cuatro tipos de espirales de prueba; las cuales se diferenciarán por los parámetros r y θ . En la tabla 1 se muestran las características de las distintas espirales a utilizar. Cabe resaltar que para cada una de las pruebas se utilizaron 20 espirales ($m = 20$) con las características (los parámetros r y θ) correspondientes a cada caso.

Tabla 1. Parámetros de las espirales de prueba.

Parámetros	Espiral1	Espiral2	Espiral3	Espiral4
Tasa de convergencia (r)	0,95	0,95	0,99	0,99
Ángulo de rotación (θ)	$\pi/4$	$\pi/2$	$\pi/4$	$\pi/2$

El objetivo de cada una de las pruebas a realizar es buscar la minimización de las cuatro funciones para tres espacios con diferente número de dimensiones ($N = 3$ y $N = 30$). Así mismo, para cada dimensión se realizará una prueba con un número máximo de iteraciones $K_{m\acute{a}x} = 100$ y $K_{m\acute{a}x} = 1000$.

A continuación, se presentan las definiciones de cada una de las funciones de prueba junto con los resultados obtenidos mediante la aplicación del algoritmo de Optimización por Espiral. Para la obtención de los resultados presentados en las tablas, se tomará el valor de *corridas* = 30, lo que significa que se realizará el procedimiento 30 veces para cada espiral y a partir de los valores obtenidos en cada corrida, se hallará la media, la desviación, el mejor valor y el peor valor.

1. FUNCIÓN SCHWEFEL

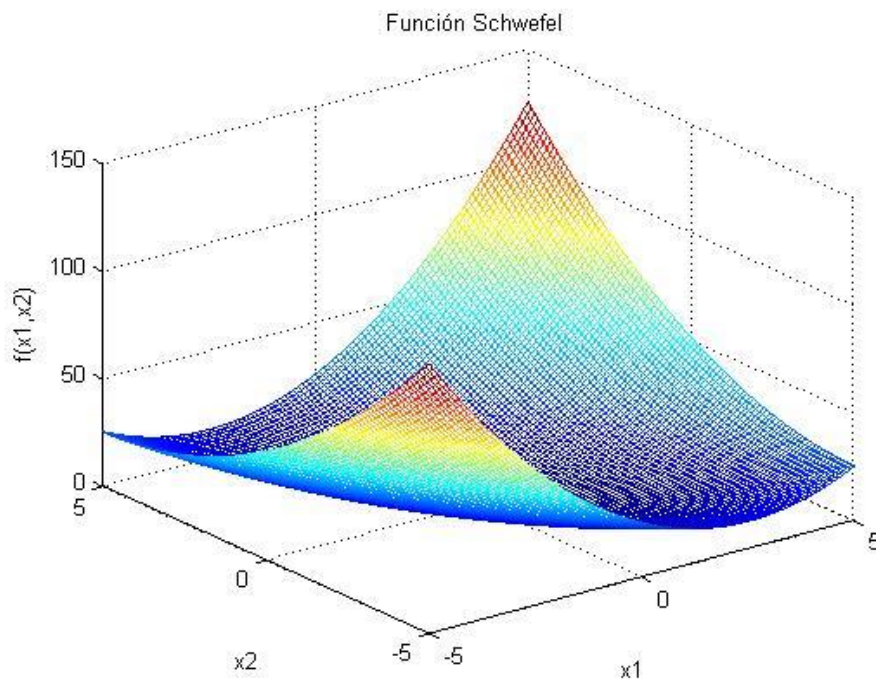
La función Schwefel está representada por la siguiente ecuación:

$$f(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2 \quad [1]$$

Ésta tiene un gran número de mínimos locales¹¹⁴ pero su mínimo global está localizado en $(0,0,0, \dots, 0)$, de forma que $f(x) = 0$.

En la Figura 1 se observa la gráfica de la función Schwefel para 2 dimensiones. Ésta cuenta con una región acotada entre $-5 \leq x_i \leq 5$ con $i = 1,2$. En la coordenada z se encuentra ubicada la variable respuesta $f(x_1, x_2)$.

Figura 1. Función Schwefel para N=2 dimensiones.



En la tabla 2 se observan los valores obtenidos mediante la aplicación del algoritmo optimización por espiral desarrollado.

¹¹⁴ SURJANOVIC, Sonja; BINGHAM, Derek. Virtual Library of Simulation Experiments: Test Functions and Datasets. Schwefel Function. [En línea]. <<<http://www.sfu.ca/~ssurjano/schwef.html>>> En: Simon Fraser University. [Citado el 15 de octubre de 2014]

Tabla 2. Resultados obtenidos para la función Schwefel

PROBLEMA	DIMENSIONES		ITERACIONES=100				ITERACIONES=1000			
			Espiral 1	Espiral 2	Espiral 3	Espiral 4	Espiral 1	Espiral 2	Espiral 3	Espiral 4
Schwefel	3	MEDIA	9,279609E-05	0,00018656	0,182424305	0,518843856	1,09534E-24	1,52442E-08	3,14971E-09	7,62788E-08
		MEJOR	7,003148E-06	1,88956E-05	0,00654528	0,149308835	6,11777E-46	7,19395E-45	1,90187E-10	1,34645E-09
		PEOR	1,227382E-03	0,020906063	0,625698763	1,073069001	0,002456635	0,00180028	3,09777E-06	0,000831355
		DESVIACIÓN	2,330499E-04	0,003783935	0,150727499	0,273197385	0,00045105	0,000417328	5,65015E-07	0,00016973
	30	MEDIA	72,67159981	11,66849065	185,7331598	83,05655991	55,04785513	8,03344474	0,072440295	2,958209213
		MEJOR	11,44438769	4,156205178	110,296922	53,27595813	14,72739428	2,921370196	7,01962E-05	1,197470882
		PEOR	218,5365431	117,397853	288,7371845	381,9568148	210,6473781	110,185253	7,363931957	26,60058058
		DESVIACIÓN	45,11786412	25,39251142	49,12750686	71,08252411	50,68692448	20,99349211	1,807002887	5,706010532

Se puede observar que cuando el número de dimensiones es 3, la mejor configuración es la presentada por la Espiral 2 ($r = 0,95$ y $\theta = \pi/2$) dado que con ésta, tanto para un número de iteraciones igual a 100 como a 1000, se obtiene la media de menor valor y la desviación más pequeña, lo que indica que la variabilidad entre los resultados es la mínima entre las cuatro opciones. Cuando el número de dimensiones aumenta a 30, para 100 iteraciones la mejor configuración es la presentada por la Espiral 2 ($r = 0,95$ y $\theta = \pi/2$), mientras que para 1000 iteraciones, la mejor es la Espiral 3 ($r = 0,99$ y $\theta = \pi/4$).

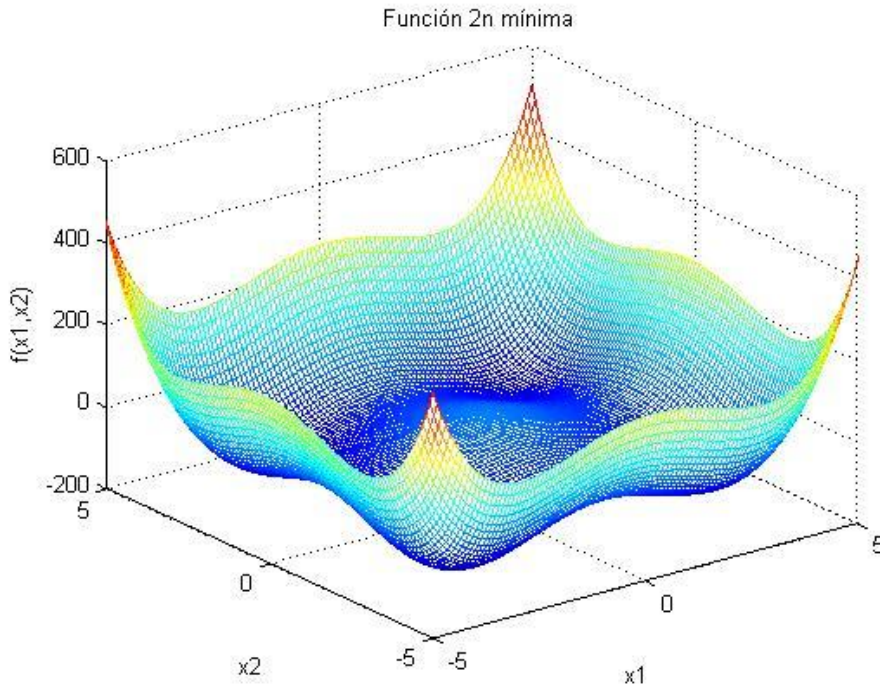
2. FUNCIÓN 2^n MÍNIMA

La función 2^n mínima presenta un mínimo global y sólo cuenta con tres mínimos locales en todo su espacio de solución. La ecuación característica de esta función está definida a continuación:

$$f(x) = \sum_{i=1}^n (x_i^4 - 16x_i^2 + 5x_i) \quad [2]$$

En la figura 2 se ilustra la gráfica de la función para 2 dimensiones. Ésta cuenta con un área de acotamiento entre $-5 \leq x_i \leq 5$ con $i = 1,2$ y la variable respuesta $f(x_1, x_2)$ se encuentra situada en la coordenada z . El mínimo de esta función se localiza en $(-2.9, -2.9, \dots, -2.9)$, lo cual ofrece un valor de $f(x) \approx -78n$. Para el caso de 2 dimensiones, $f(x) \approx -156$.

Figura 2. Función 2n Mínima para N=2 dimensiones.



En la siguiente tabla se pueden observar los valores obtenidos mediante la aplicación del algoritmo optimización por espiral.

Tabla 3. Resultados obtenidos para la función 2n Mínima

PROBLEMA	DIMENSIONES		ITERACIONES=100				ITERACIONES=1000			
			Espiral 1	Espiral 2	Espiral 3	Espiral 4	Espiral 1	Espiral 2	Espiral 3	Espiral 4
2n Mínima	3	MEDIA	-233,7281335	-206,7196262	-224,6472359	-201,13699	-234,8736213	-222,7929154	-234,9969939	-206,7235558
		MEJOR	-234,9964954	-234,9959233	-234,3165172	-234,0228012	-234,9969942	-234,9969942	-234,9969942	-234,9969942
		PEOR	-200	-177,8984748	-200	-157,8536247	-200	-178,1952911	-197,5564395	-178,2450691
		DESVIACIÓN	13,44582244	16,4815635	9,746506009	22,65306875	12,80999616	15,48176523	11,93540644	15,36067573
	30	MEDIA	-1841,910004	-1856,539163	-1085,271861	-1248,138825	-1888,95773	-1845,137929	-2038,962058	-1992,033259
		MEJOR	-2015,861718	-2045,293514	-1374,753132	-1412,719449	-2106,283755	-2006,966912	-2208,602704	-2150,864723
		PEOR	-200	-200	-200	-200	-200	-200	-200	-200
		DESVIACIÓN	90,41262494	77,44584679	170,6744956	64,40767893	100,7315849	80,24465594	75,75044988	77,64769179

Para este caso, cuando las dimensiones son 3 y el número de iteraciones son 100, se observa que la Espiral 1 ($r = 0,95$ y $\theta = \pi/4$) es la que ofrece los mejores

resultado, esto teniendo en cuenta que el mínimo sería $f(x) \approx -78(3) \approx -234$; la Espiral 1, a pesar de no tener la desviación más pequeña, es aquella que presenta la media más cercana al valor teórico de forma que el intervalo de valores que ofrece esta configuración ubica al valor teórico en la parte central del mismo mientras que la Espiral 3 ($r = 0,99$ y $\theta = \pi/4$), la cual es la que tiene la menor desviación, presenta una media más alejada del valor teórico obteniendo un intervalo de valores que ubica dicho valor cerca del límite superior del mismo. Para un número de iteraciones igual a 1000, tanto la Espiral 1 ($r = 0,95$ y $\theta = \pi/4$) como la Espiral 3 ($r = 0,99$ y $\theta = \pi/4$) ofrecen valores de media muy cercanos al valor teórico pero la Espiral 3 cuenta con la desviación más pequeña entre las cuatro opciones. Cuando el número de dimensiones aumenta a 30, para un número de iteraciones igual a 100, la mejor configuración es la ofrecida por la Espiral 2 ($r = 0,95$ y $\theta = \pi/2$), la cual ofrece el valor medio más cercano al valor teórico ($f(x) \approx -78(30) \approx -2340$) y cuenta con la menor desviación; para 1000 iteraciones, la Espiral 3 ($r = 0,99$ y $\theta = \pi/4$) es la que ofrece la menor desviación y da la media más cercana al teórico.

3. FUNCIÓN RASTRIGIN

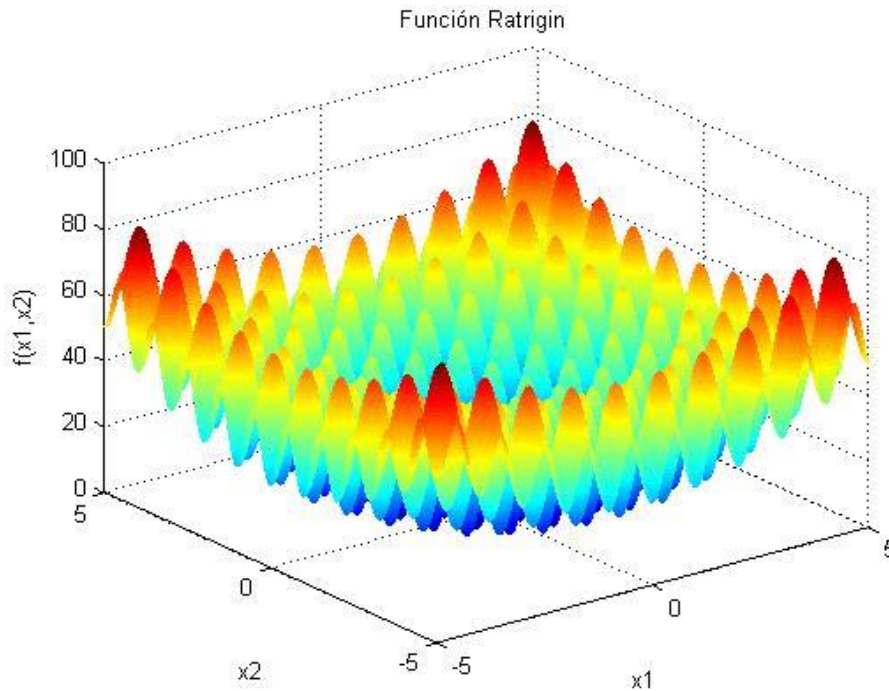
La función Rastrigin es una de las funciones más usada a la hora de evaluar algoritmos; ésta consta de muchos mínimos locales pero sólo tiene un mínimo global ubicado en $(0,0,0, \dots, 0)$ ¹¹⁵ obteniendo un $f(x) = 0$. La siguiente ecuación representa la función Rastrigin para n variables.

$$f(x) = \sum_{i=1}^n (x_i^2 - 10 \cos 2\pi x_i + 10) \quad [3]$$

La figura 3 muestra la gráfica de la función para 2 dimensiones. Su área de acotamiento se encuentre entre $-5 \leq x_i \leq 5$ con $i = 1,2$ y la variable respuesta $f(x_1, x_2)$ está ubicada en la coordenada z .

¹¹⁵ Mathworks. Documentation. Minimize Rastrigin's Function. [En línea] <<<http://www.mathworks.com/help/gads/example-rastrigins-function.html>>> [Citado el 15 de octubre de 2014]

Figura 3. Función Rastrigin para N=2 dimensiones.



Los resultados obtenidos para la función Rastrigin mediante la aplicación del algoritmo desarrollado se muestran en la siguiente tabla.

Tabla 4. Resultados obtenidos para la función Rastrigin

PROBLEMA	DIMENSIONES		ITERACIONES=100				ITERACIONES=1000			
			Espiral 1	Espiral 2	Espiral 3	Espiral 4	Espiral 1	Espiral 2	Espiral 3	Espiral 4
Rastrigin	3	MEDIA	1,20065973	1,102444176	3,864686882	8,425505492	1,989918114	1,989918114	0,994959635	1,989922545
		MEJOR	0,007858694	0,017993063	1,06215896	2,743941059	0	0	1,71529E-07	9,23501E-07
		PEOR	5,22666332	5,012686145	6,874745732	13,20353216	4,974790248	5,969749305	1,989983488	7,959670474
		DESVIACIÓN	1,231963553	1,143364721	1,531699939	2,424875961	1,277040841	1,763827901	0,797880281	1,954110946
	30	MEDIA	195,3600839	107,3821906	379,8118525	273,6213745	191,2389212	75,35085256	145,7609838	56,72950274
		MEJOR	129,9264126	57,19160818	331,7722143	229,4370804	114,5767942	42,37481011	68,65231725	22,59755907
		PEOR	272,068067	172,7521382	436,4126929	293,784595	251,8612037	141,9503095	256,6983159	157,3631773
		DESVIACIÓN	33,68316358	27,56374217	28,18734134	14,31517874	37,87183447	23,09031011	46,59649588	32,26930654

Se puede observar que cuando el número de iteraciones es igual a 100, tanto para 3 dimensiones como para 30 dimensiones, la mejor configuración es la

presentada por la Espiral 2 ($r = 0,95$ y $\theta = \pi/2$), dado que es ésta la que cuenta con el menor valor de media y el menor valor de desviación. Cuando el número de iteraciones aumenta a 1000, para un número de dimensiones igual a 3, la mejor Espiral es la 3 ($r = 0,99$ y $\theta = \pi/4$), la cual ofrece la media más cercana al mínimo global y tiene el menor valor de la desviación. Para 30 dimensiones, la mejor es la Espiral 2 ($r = 0,95$ y $\theta = \pi/2$), que a pesar de no tener el valor de la media más cercano al mínimo global, tiene el valor de la desviación más pequeño y, se puede observar que el peor valor obtenido por esta configuración es el más cercano al mínimo global entre las cuatro configuraciones.

4. FUNCIÓN GRIEWANK

La función Griewank tiene un mínimo global y muchos mínimos locales que se encuentran distribuidos de forma regular en toda la región; esta cantidad de mínimos locales aumentan las posibilidades de que el algoritmo se quede encerrado en estos y no pueda encontrar el mínimo global que se encuentra ubicado en $(0,0,0,\dots,0)$ ¹¹⁶ con un valor de $f(x) = 0$. Dicha función es representada por la siguiente ecuación:

$$f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad [4]$$

En la figura 4 se observa la gráfica de la función para 2 dimensiones en donde la variable respuesta $f(x_1, x_2)$ se encuentra en la coordenada z . En este caso la función se encuentra acotada entre $-50 \leq x_i \leq 50$ con $i = 1, 2$.

¹¹⁶ SURJANOVIC, Sonja; BINGHAM, Derek. Virtual Library of Simulation Experiments: Test Functions and Datasets. Griewank Function. [En línea] <<<http://www.sfu.ca/~ssurjano/griewank.html>>> En: Simon Fraser University. [Citado el 15 de octubre de 2014]

Figura 4. Función Griewank para N=2 dimensiones.

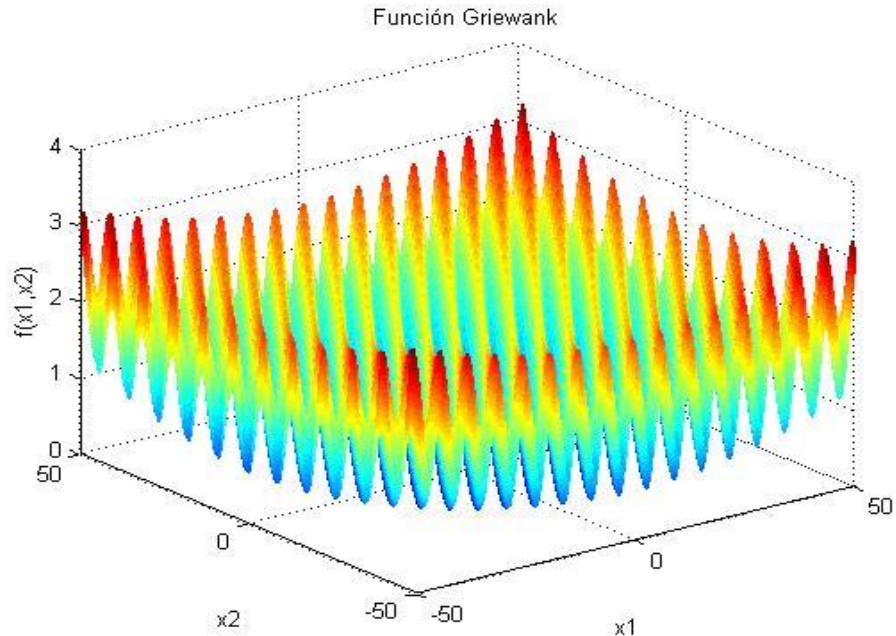


Tabla 5. Resultados obtenidos para la función Griewank.

PROBLEMA	DIMENSIONES		ITERACIONES=100				ITERACIONES=1000			
			Espiral 1	Espiral 2	Espiral 3	Espiral 4	Espiral 1	Espiral 2	Espiral 3	Espiral 4
Griewank	3	MEDIA	0,063583647	0,050011538	0,113523747	0,16782023	0,061624639	0,045594552	0,020952642	0,044240161
		MEJOR	0,001049305	0,010599253	0,024084213	0,058832893	0	0,00739604	0,007396079	0,00739632
		PEOR	0,200359957	0,144165252	0,250121627	0,300824536	0,204689479	0,187354463	0,059178277	0,187856198
		DESVIACIÓN	0,047648586	0,032629824	0,05281675	0,058092447	0,045130238	0,039508029	0,015134944	0,036072615
	30	MEDIA	1,282455982	1,039206219	3,315983785	2,037551225	1,324327286	1,001455346	0,012329447	0,155920154
		MEJOR	1,053480062	0,735479708	2,609448403	1,873460339	0,940079225	0,628916737	6,5659E-06	0,005162455
		PEOR	2,123186812	1,124990495	4,56082589	2,26830567	1,861227804	1,121583242	0,054111928	0,988192553
		DESVIACIÓN	0,299416814	0,078884436	0,482300402	0,076904335	0,224322821	0,138092045	0,013537114	0,223811278

Para este último caso, cuando el número de iteraciones es equivalente a 100, tanto para 3 dimensiones como para 30 dimensiones, la mejor configuración es la presentada por la Espiral 2 ($r = 0,95$ y $\theta = \pi/2$), ofreciendo tanto la desviación

más pequeña como el valor medio más cercano al mínimo global. Para 1000 iteraciones, la mejor espiral es la 3 ($r = 0,99$ y $\theta = \pi/4$), quien ofrece la menor desviación y la mejor media entre las cuatro opciones.